STRAPDOWN ANALYTICS

Second Edition

PART 1

Paul G. Savage

Published By:

Strapdown Associates, Inc.
Maple Plain, Minnesota

First Published 2000

ISBN: 0-9717786-0-4
(Vol. 1 of 2 Vol. Set)

Foreword

This two volume text provides a detailed comprehensive discourse on the analytics of strapdown inertial navigation systems (INS's), the basic technology used on modern day commercial and military aircraft, guided missiles, surface ships and underwater vehicles. Based on his first-hand experience in this field, the author has provided a unique service to the aerospace industry in preparing this technical dissertation on the algorithms implemented in the strapdown system computer, and the analytics (and software) associated with system software validation, system test, simulation, performance analysis, and the analytical design methodology used in deriving the strapdown equations. Included is an in-depth chapter dealing with Kalman filter theory and its application to the aiding of a strapdown INS.

Strapdown Analytics has been prepared for the reader who may not have had experience in navigation or Kalman filtering. The analytical material presented is derived from scratch, showing the developmental steps in rigorous detail, without relying on reference material for supporting analytics. The book is complicated, yet complete and understandable by analytically inclined graduate students and practicing engineers. The book can be viewed as the text for an advanced course one might take following the introductory course taught by the author, *Introduction To Strapdown Inertial Navigation Systems*.

This is the second edition of *Strapdown Analytics*. It contains all material provided in the first edition including errata corrections uncovered since the original publication in 2000. This edition contains an additional Chapter 19 which presents three relevant strapdown papers published by the author since 2000.

About The Author

Paul G. Savage is an internationally recognized expert in the design and test of ring laser gyro strapdown inertial navigation systems. He is the president of Strapdown Associates, Inc. (SAI), a Minnesota engineering company he founded in 1980 to further the advancement of strapdown inertial technology. Since the down-sizing of SAI in 1995, Mr. Savage has continued to provide inertial navigation consultation services and teach his *Introduction To Strapdown Inertial Navigation Systems* course to the aerospace industry. He also serves as a reviewer for the AIAA Journal of Guidance, Control, and Dynamics.

From 1980 to 1995, SAI, under Mr. Savage's leadership, provided technical engineering development and educational support for system configuration definition, flight software development, system simulation, and testing for strapdown inertial system development programs for military and commercial aircraft, land vehicles, cruise missiles, and ground and air-launched tactical missiles. Strapdown Associates has held contracts in these areas with Naval Weapons Center, Naval Air Development Center, McDonnell Douglas, Honeywell, Sundstrand, Systron Donner, Texas Instruments, Rockwell International, Aerojet, Contraves, Bendix, Northrop, General Dynamics, Boeing, Oklahoma City Air Logistics Command / Tinker Air Force Base, Environmental Research Institute of Michigan, ARINC Research, Baker Hughes, Alliant Techsytems, Lockheed Martin and United Defense.

From 1963 to 1980, Mr. Savage was employed at Honeywell Avionics Division as Senior Principal Engineering Fellow where he led engineering design teams and provided technical consultation to Honeywell engineering managers for system design, analysis, software development, simulation, and integration/test in the evolutionary development of laser gyro strapdown inertial navigation systems for military and commercial aircraft. From 1971 through 1975, Mr. Savage was the engineering manager and system design engineer for the Honeywell LINS-0 strapdown inertial system, the first to prove the readiness of laser gyro strapdown inertial navigation technology for 1 nautical-mile-per-hour accuracy aircraft applications as demonstrated during a landmark flight test series at Holloman Air Force Base in 1975.

Mr. Savage is a graduate from the Massachusetts Institute of Technology where he received his MS and BS degrees in Aeronautical Engineering in 1960.

Preface

Strapdown inertial navigation is a technology for autonomously determining position location using a portable system containing strapdown inertial angular rate sensors, acceleration sensors, and a data processing computer. "Strapdown" refers to the method of mounting the inertial instruments directly to the user vehicle, rather than on a gimbaled platform used to isolate the sensors from vehicle rotation in earlier inertial systems. Inertial navigation systems (often with additional supporting navigation aids - e.g., positioning data from Global Positioning System (GPS) receivers) are utilized extensively as the basic navigational element on military and commercial aircraft, guided missiles, surface ships and underwater vehicles. Advances in inertial sensor and computer technology in the 1970's made it possible to achieve accuracies with strapdown systems that equaled earlier gimbaled technology performance for lower cost and improved reliability. By year 2000, virtually all inertial navigation systems had become (or were being retrofitted to) the strapdown type (an exception being ballistic missile guidance systems which continued to use gimbaled technology).

The title of this book, *Strapdown Analytics*, has a double meaning. It deals with the analytics associated with strapdown inertial navigation systems; it is also a detailed tutorial treatise describing the analytics used at my company, Strapdown Associates, Inc. (SAI or "Strapdown" for short), over the 20 year period following its inception in 1980. During this time, SAI developed strapdown inertial navigation application software in both free-inertial and Kalman-filter-aided configurations for various user groups including validation software, system test software, simulators, and associated performance analyses. In addition to being SAI's president, one of my principal responsibilities at SAI was to prepare the technical documentation for SAI programs. Since 1981, I have also provided an introductory course to the general public, *Introduction To Strapdown Inertial Navigation Systems*, covering the various strapdown system hardware, software, inertial sensor, and system technological elements.

While teaching the Strapdown Inertial course, I have been frequently asked for references covering the course topics in more detail (in addition to the handout material provided to course attendees). While I have been able to recommend several books covering inertial sensors and Kalman filtering, I have been at a loss to recommend comprehensive documents providing a detailed account of the analytical aspects of strapdown inertial navigation systems. At best, I have referred to some papers that have been published over the years by several authors in the trade journals and at technical symposiums. For SAI customers, I have referenced the documentation provided with SAI's software package deliverables, which though fairly comprehensive for their particular functions, are not in the form of tutorial text material, and presume a certain level of basic strapdown analytical background by the reader.

This book represents a detailed comprehensive tutorial account of my knowledge in the analytical aspects of strapdown inertial navigation. This includes not only the algorithms implemented in the strapdown system computer, but the analytics (and software) associated with system software validation, system test, simulation, and system performance analysis. It also implicitly illustrates the analytical design methodology I have used in deriving strapdown equations. The book was prepared using SAI independently developed technical documentation and my personal notes as a starting base, and then rewritten for tutorial clarity, expanded to fill in technical voids in equation derivations, modified for technical parameter compatibility and cross-referencing between sections, enhanced for analytical improvements uncovered during book preparation, and expanded to cover important material that I had not previously documented. Beginning with 1500 pages from assorted SAI documents, I had hoped to develop a text book with a reasonable page count (e.g., 800 pages). Due to the added material and ground-rules I set during book preparation (discussed below), the final version was more than 1500 pages, requiring two volumes.

In preparing this book, I set several goals to achieve what I believe is required of a textbook for easy reader comprehension. Many of the goals were selected to avoid problems I have experienced in the past when using analytical text documents. The unfortunate penalty was increased page count which I decided at the onset to be worth the cost. The goals were as follows:

- Clearly delineated parameter definitions separated from the main text, including a parameter index for referencing back to the book location where the parameter was defined.

- Repeating a parameter definition where needed for clarity in sections that are far removed from the section in which it was originally defined.

- Deriving equations beginning from basics; avoiding the practice of referencing supporting equations to other source material. The entire book contains only 40 references, some of which are the basic textbooks I used at college. One of the reasons for additional supporting documentation has been to substantiate analytical results obtained. In this book, equations are derived from scratch including supporting analytics. As such, results are self-verifying without supporting documentation (with some minor exceptions, in which case supporting documents are referenced; e.g., standard gravity models in References 3 and 4). My analytical experience in the aerospace industry is the principal reference for the analytical derivations provided, supplemented with theoretical background material provided by textbooks I have referenced in the book (notably, Reference 6 for Kalman filter theory and Reference 3 for vector/coordinate-frame nomenclature).

- Avoiding reference material that was not used in the book preparation. Exceptions are the selected background material on attitude integration algorithms referenced in Chapter 7, an important part of strapdown analytical development history, and References 16 and 21, fairly recent textbooks on inertial sensors (as of year 2000). I decided at the onset that I would not perform a literature search to find additional background material that readers might find useful. This book is based on my knowledge base acquired during 39 years of experience in the aerospace industry, i.e., the material I have used on a daily basis in performing my engineering assignments.

- Providing equation derivations that show the intermediate steps to avoid having the reader accept results on faith, or spend valuable time filling in the voids for verification.

- Developing equations in full closed-form wherever possible without reverting to the use of linearization approximations (an obvious exception is Kalman filter error state modeling which is inherently based on linearization techniques). Much of the analytical results developed in this book are designed for translation into software on a host computer. Advancements in computer technology make it possible to implement the added closed-form equation computational burden for virtually no cost penalty. The important benefit is the elimination of linearization process error and associated documentation/validation requirements for assuring sufficient accuracy.

- Deriving equations in vector format when possible for more generality, and in this case, to reduce page count.

This book should be viewed as the text one might use for an advanced course on strapdown inertial navigation, e.g., the course that might follow my introductory strapdown course. It is not a simple book, covering some very complicated analytical subjects. Nevertheless, it has been designed for comprehension by engineering analysts or graduate students having only a basic working knowledge of vector calculus and matrix operations. Previous background in inertial navigation or navigation in general is not necessarily required. However, it is helpful.

One final word of caution. Although every effort has been made to eliminate errors in the book (including careful proofing of five technical draft copies and the final print copy), it is inevitable that some errors have passed undetected (e.g., typographical errors in the equations themselves or in the equation number cross-references used liberally throughout the book). As such, if readers plan to use the book material for important engineering projects, they are encouraged to re-verify the material by analytical means and simulation.

Good luck.

Paul G. Savage

Acknowledgments

An important part of the writing of any book is the proof reading. In the case of an analytical text book this is particularly important because of the many possible areas where errors can go undetected (e.g., incorrect symbol, misplaced or missing bracket, wrong Font, missing vector notation, flawed equation derivation, assumed analytic characteristic that was never proven, incoherent explanation, incorrect equation or section numbers in the text and in cross-referencing). It is virtually impossible for the writer of such a book to find and correct all of the possible errors. Writers are so familiar with their own work that they are blind to many of their errors during the proof-reading process. To prevent this problem, another reader (or several) typically assist in the proofing process.

The proof reader of an analytical text book should have the basic technical ability to understand the equations presented and their analytical derivations. Ideally, the proof reader should have had no previous background in the text book material to avoid introducing inadvertent technical prejudices in the reviewing process. For the proofing process itself, the ideal proof reader would independently rederive all equations in the book to verify their accuracy and analytical cross-references, and provide comments to the author regarding understandability of the written text and equation derivation methods. An important implicit requirement in this regard is confidence on the part of the author that the proof reader will perform the task thoroughly and dependably with the same concern that the reader has in the accuracy of the final result.

For the *Strapdown Analytics* book, I have been fortunate to have found someone with superb technical credentials, and who was willing to provide such a dedicated proof-reading service; my daughter Kelly M. Roscoe. Kelly received Bachelor and Master of Science Electrical Engineering degrees with honors from the Massachusetts Institute of Technology (MIT), and has had technical experience in military system design/management at Rockwell Seal Beach Facility and advanced optical materials process control at Minnesota Mining & Manufacturing. I have Kelly to thank for the accuracy and understandability of the final published version of *Strapdown Analytics*. Without her help, the book in its present form would not have been possible.

I would also like to express my gratitude to my wife Paula for her encouragement during the four year period that this book was written, and for her direct assistance in the final formatting and proofing stages of book preparation. Her understanding and patience when my thoughts were preoccupied with writing was most appreciated.

Lastly, I would like to express my acknowledgment to the Massachusetts Institute of Technology for my basic engineering education, Lockheed Missiles & Space Company and Honeywell, Inc. for exposure as an employee to the general business of inertial navigation, to employees and customers of my company Strapdown Associates, Inc. (SAI) who made it possible for SAI to exist, to past attendees of SAI's strapdown inertial navigation course, and to the technical colleagues I have informally known over the years. Involvement with these people and associated programs during my career provided invaluable experience and the environment for developing the technical expertise needed to prepare this book.

Paul G. Savage

Contents For Parts 1 And 2

Part 1

Part 2

---

1 Introduction

Inertial navigation is an autonomous process of computing position location by doubly integrating the acceleration of a point whose position is to be determined. The fundamental concept is illustrated in Figure 1-1.

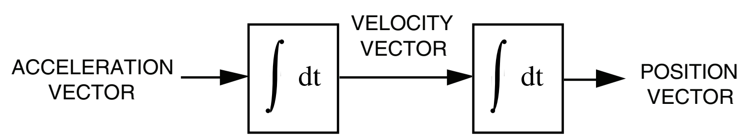


Figure 1-1 Fundamental Inertial Navigation Concept

Figure 1-1 shows a three-dimensional acceleration vector being integrated once to determine a three-dimensional velocity vector and again to obtain a three-dimensional position vector. Also implicitly represented in Figure 1-1 is the requirement to initialize the velocity/position integrators prior to the start of inertial navigation. In general, the initialization process requires knowledge of starting velocity and position.

An inertial navigation system (INS) implements the Figure 1-1 concept using a cluster of accelerometers to sense the acceleration vector components and a digital computer to perform the integration operations. The direction of the accelerometers (and the associated acceleration vector) is determined using a cluster of angle or angular rotation sensing instruments (e.g., gyros) that are physically mounted in a known geometrical relationship relative to the accelerometers. To ideally implement Figure 1-1 in the INS, the accelerometers would be specified to provide measurements of total acceleration (i.e., the second derivative of position). In general, total acceleration is composed of two fundamental parts: gravity acceleration created by the gravity field surrounding the INS, and "specific force" acceleration produced by forces acting on the vehicle containing the INS (which through mechanical linkage, produce forces within the INS accelerometers). Due to basic limitations of fundamental physics, accelerometers can only be designed to measure the specific force component of acceleration. Hence, to determine total acceleration for Figure 1-1, the gravity acceleration must be added to the accelerometer measurements. The result is the fundamental inertial navigation system concept depicted in Figure 1-2.

Figure 1-2 shows the gravity acceleration being calculated as a function of INS computed position. The gravity calculation is performed within the INS computer. Implied by this operation is a computerized model of the gravity vector field as a function of position in the space in which the INS is to be operated. The INS is mounted within a "user vehicle" whose position is to be calculated by the INS. Thus, by calculating its own position, the INS also determines the position of the vehicle in which it is mounted. Two classical INS mechanization approaches have been utilized to generate the specific force acceleration vector from the accelerometers (i.e., vector components and direction): the "gimbaled" approach and the "strapdown" approach.

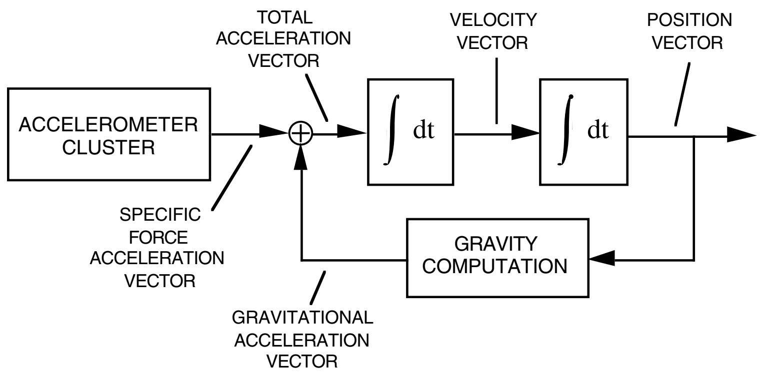


Figure 1-2 Fundamental Inertial Navigation System Concept

In the gimbaled approach, the accelerometers are mounted to a rigid structure that is mechanically coupled to the user vehicle by a set of concentric gimbals. The gimbals are connected to the accelerometer mount, to each other, and to the user vehicle by bearing assemblies that provide rotational freedom around the bearing axes. The "gimbaled platform" concept is depicted in Figure 1-3.

In Figure 1-3, three accelerometers (the cube structures) are mounted to the inner platform with their input sensing axes orthogonal. The inner platform angular orientation is controlled by electric torque motors mounted around the gimbal bearing axes. The control signals for the gimbal torque motors are provided by inertial angular rotation sensing instruments (gyros) mounted to the inner platform (cylindrical structures) with input sensing axes (dashed lines) orthogonal. By controlling the gyro outputs to be zero through the resulting gimbal torque motor closed-loop servo action, the inner platform (with its accelerometers) is controlled to maintain a specified angular orientation. To make the platform rotate at a prescribed angular rate (selected by the INS computer), the platform gyros are electrically biased by computer specified platform rotation rates, using biasing elements contained within each gyro. The platform gyro outputs are proportional to the integrated difference between the angular rotation rate dynamically input to the gyros (about their input axes) and the electrically applied gyro bias inputs. The gimbal torque motor control loops maintain the gyro outputs at zero, hence, the dynamic angular rate into the gyros (i.e., the inner platform angular rate) is forced to balance the gyro electrical biasing rate. The net effect is that the inner platform (and the accelerometers) are controlled to match the INS computer software specified integrated angular rate orientation profile (known in the INS computer), hence, the angular orientation of the accelerometers becomes implicitly known in the INS computer. The computer is then able to define the specific force vector using the accelerometer outputs for the vector component values, and the known orientation of the inner platform (the "inertial sensor platform") for the vector direction.

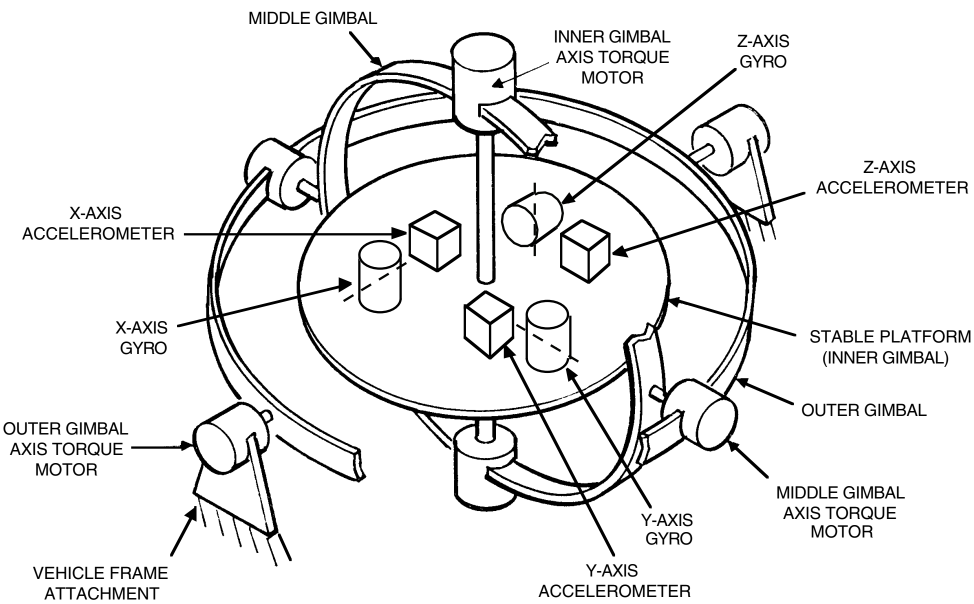


Figure 1-3 Gimbaled Inertial Platform Concept

A complete gimbaled INS consists of the Figure 1-3 "gimbaled platform", the INS computer, and associated electronics, all contained within a common chassis. The INS chassis is then physically connected to the user vehicle using a rigid "INS mount" assembly. Figure 1-4 illustrates the gimbaled platform interfaced to a computer in a gimbaled inertial navigation system. The navigation computation block in Figure 1-4 performs the Figure 1-2 integration and gravity computation operations.

In the strapdown approach, the interconnecting gimbal structure of Figure 1-3 is eliminated, and the inertial sensor platform (containing the inertial sensors) is mounted directly within the INS chassis (i.e., "strapped down" to the INS and to the user vehicle, thus the name "strapdown" to describe the technology). To perform the accelerometer orientation determination function (provided mechanically by the gimbal assembly in Figure 1-3), the

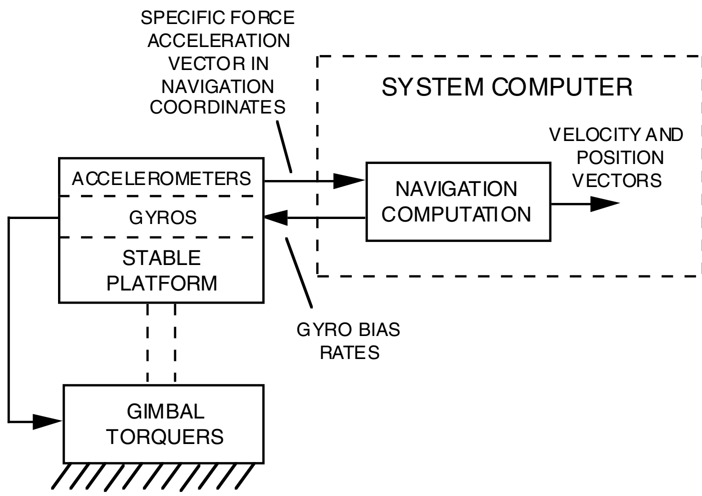


Figure 1-4 Gimbaled Inertial Navigation System

strapdown INS calculates the orientation of the strapdown accelerometers by processing a sensor assembly angular rate vector measured by the strapdown angular rate sensors (i.e., the so-called "body rate" signals) in the INS computer. (Henceforth, we will use the term "angular rate sensor" to generically define inertial sensors that sense angular rate. The more commonly used "gyro" term refers to inertial sensing instruments based on gyroscopic rotating mass dynamic principles. Modern day strapdown inertial sensors are based primarily on optical or Coriolis vibrating mass principles, hence, technically should not be called gyros, even though they measure the same input sensed by classical spinning mass gyroscopes. Spinning mass gyroscopes are also angular rate sensors, either directly, or in an integral sense). Figure 1-5 illustrates the strapdown INS for comparison with the Figure 1-4 gimbaled INS.

Both the strapdown and gimbaled system concepts in Figures 1-4 and 1-5 provide the same specific force acceleration vector inputs to the velocity/position integration navigation computation software in the system computer (i.e., the Figure 1-2 specific force acceleration vector). In the gimbaled system, the specific force vector is measured directly by the accelerometers on the gimbaled platform whose orientation (in the form of a "navigation coordinate frame" attitude) is selected (and controlled) by the navigation computer software. The resulting specific force vector in navigation coordinates is then processed as in Figure 1-2 to determine velocity and position. In the strapdown system, the specific force acceleration is first measured by the strapdown accelerometers as a vector in a "strapdown sensor coordinate frame", and is then analytically rotated (by the INS computer software) from the strapdown sensor coordinate frame into the navigation coordinate frame. The result is the specific force vector in navigation coordinates used in Figure 1-2 for integration into velocity/position. To perform the coordinate frame rotation operation (called a "vector transformation"), the angular orientation between the strapdown sensor and navigation coordinate frames must be known in the system computer. It is found by a software integration operation using sensor coordinate frame angular rates measured by the strapdown angular rate sensors, and navigation coordinate frame angular rates specified by the INS software. The navigation coordinate frame angular rates are the same signals used in Figure 1-4 to bias the angular rate sensors in the gimbaled platform. Thus, both the strapdown and gimbaled systems generate the same navigation coordinate frame version of the specific force vector (for the Figure 1-2 input) and both use the same navigation frame angular rates in finding (or controlling) the specific force vector component coordinate frame.

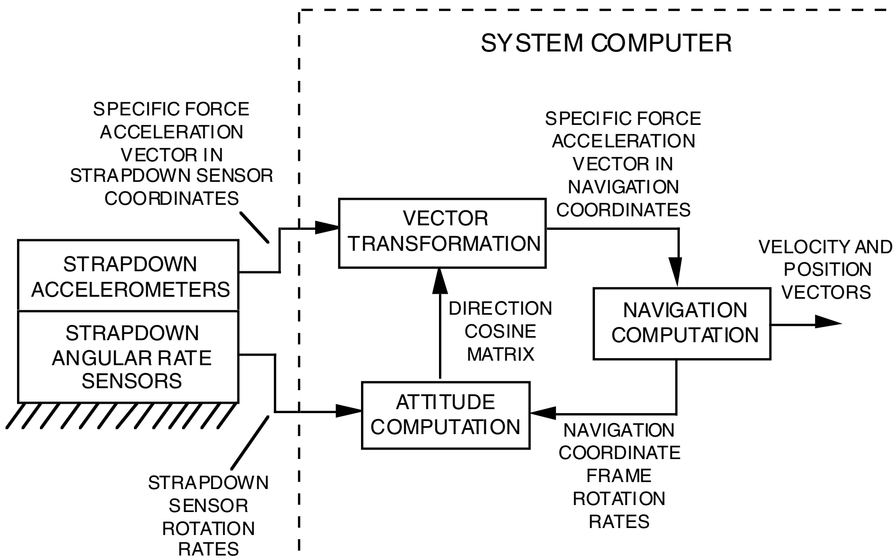


Figure 1-5 Strapdown Inertial Navigation System

In a general sense, the difference between a strapdown and a gimbaled system can be considered as a tradeoff between mechanical complexity (for the gimbaled system) versus computational complexity (for the strapdown system). From a performance standpoint, a fundamental handicap for the strapdown system is that the strapdown sensors (particularly the angular rate sensors) are exposed to the full vehicle angular rotation rate, whereas for the gimbaled system, the inertial sensor platform rate is controlled to be small, independent of vehicle angular rate. Meeting specified angular rate sensor accuracy requirements under high dynamic vehicle angular rate inputs (i.e., for the strapdown system) is generally more difficult to achieve than for the low benign angular rate environment of the sensors in a gimbaled platform. In fact the basic gimbaled platform concept was originated as a means of shielding spinning wheel gyros from vehicle angular rates, thereby making it possible to design gyros that would meet system accuracy requirements. With the advent of the ring laser gyro in the mid 1970's (an angular rate sensor based on optical rather than spinning mass dynamic principles), it became possible to achieve high accuracy under high angular rates. During the same time period, advancements in computer technology made it possible to implement the added strapdown computational burden for virtually no production cost penalty. The merging of these two technologies initiated the conversion of inertial navigation from original gimbaled to modern day strapdown technology. With few exceptions, the conversion process was complete by the year 2000. Today, strapdown inertial navigation and inertial navigation are synonymous.

The technologies utilized in the design of modern strapdown inertial navigation systems include inertial sensors (angular rate sensors and accelerometers), electronics (digital and analog), software, mechanical and thermal design, testing, and associated analytics. This book, in two parts, deals only with the analytical aspects of strapdown inertial navigation for the software resident in the navigation computer, software for system testing, and system performance analysis. For information on strapdown inertial sensors, the reader is referred to available literature on the subject (e.g., References 16, 21, 31 and 32), and is encouraged to contact the manufacturers of particular inertial components for further detail. Similarly, the reader should contact the manufacturers of inertial navigation systems for particulars on currently available system technologies. The remainder of this Chapter 1 Introduction describes the analytical material covered in each chapter of the book.

Chapter 2 provides a comprehensive guide to the terminology used throughout the book including mathematical notation, coordinate frame definitions and parameter definitions. Due to the diversity of analytical topics covered, it became virtually impossible to adopt a single meaning for each parameter and coordinate frame used throughout the book. To circumvent this difficulty, a Parameter Index and Coordinate Frame index is provided in the back of each of the Part 1 and Part 2 book volumes (in addition to the Subject Index) to facilitate locating parameter/coordinate frame definitions in the main text. The parameters and coordinate frames are alphabetically listed in these indexes with the equation number preceding their definition in the main text. In addition, a listing of mathematical symbols used is provided in Section 2.1 of Chapter 2, also with the equation number preceding their definition in the main text. To facilitate the recall of parameter, coordinate frame, and mathematical symbol definitions, they are separated and indented from each paragraph throughout the book, and repeated in sections that are far separated from sections in which they were first defined. The overall intent is to avoid the problem readers have found with many textbooks of forgetting the meaning of a particular variable and having to spend frustrating time trying to find its definition buried in the main text.

Chapter 3 provides an introduction to the basic mathematics utilized throughout the book including vector operations in selected coordinate frames, their analytical conversion process between coordinate frames, their component rates of change in rotating coordinates, and basic analytical operations for describing coordinate frame angular orientation ("attitude"). Attitude parameters discussed are the direction cosine matrix, the rotation vector, Euler angles, and the attitude reference quaternion, including analytical equivalencies between the parameters, and the analytics used to describe their rates of change. The concluding section provides a detailed discussion on methods for describing attitude and vector error characteristics. For the attitude error discussion, error angle vectors are derived describing the orientation error in the relative attitude between two coordinate frames, considering one of the frames as the reference and the other as having the orientation error. For the velocity error discussion, velocity error equations are developed as a function of the coordinate frame in which the velocity error is defined and the coordinate frame in which the error is to be evaluated.

Chapter 4 uses the Chapter 3 analytics to develop the equations that would typically be implemented in an earth based strapdown inertial navigation system computer for calculating attitude, velocity and position (as in Figure 1-5). The attitude/velocity/position calculations are analytically described in the form of time rate differential equations that would be continuously integrated in the INS computer using suitable digital integration algorithms. For the attitude determination function, both direction cosine matrix and quaternion forms are presented. Velocity is defined as INS position rate relative to the earth. The velocity vector rate equation is developed for integration in a locally level "navigation coordinate frame" (e.g., of the azimuth wander or free azimuth type, both of which are described), and includes the effect of navigation coordinate frame and earth's angular rotation rate relative to non-rotating inertial space. The attitude rate equations are derived to relate the strapdown sensor coordinate frame to the locally level navigation frame. Strapdown acceleration transformation operations are included for converting the accelerometer measured specific force acceleration into its navigation frame equivalent, the specific force input to the velocity rate equation. The position rate equation is defined in two parts: altitude rate and the rate of change of a direction cosine matrix relating navigation coordinates to a specified earth fixed coordinate frame ("position direction cosine matrix"). Included in the altitude rate equation is a method for controlling vertical error build-up in velocity and position ("vertical channel divergence") using an input pressure altitude signal. Equations are developed for converting the computed attitude/velocity/position data to equivalent output formats (e.g., roll/pitch/heading Euler angle attitude, north/east/vertical velocity components, latitude/longitude/altitude position or a position vector from a selected earth fixed position location to the INS). Equivalency equations are also provided for converting one form of position representation to another. Chapter 4 includes a brief discussion on initialization requirements covered in detail in Chapter 6. At the conclusion of Chapter 4, a summary table is provided listing the principal Chapter 4 equations and the inputs required from other sections of the book for earth related parameters and initialization operations.

As part of the inertial navigation software, analytical models must be included describing gravity in the space potentially occupied by the INS and to describe the referencing surface for position definition (e.g., the surface of the earth including its rotation rate relative to inertial space). Chapter 5 analytically describes the earth in terms of its classically represented ellipsoidal reference surface (approximately at mean sea level), and the analytical definition of earth referenced parameters used in the INS computer (e.g., latitude, longitude, altitude above the earth reference ellipsoid, the relationship between INS horizontal velocity and the angular rate of the locally level navigation coordinate frame (called "transport rate"), and radii of curvature of the earth's surface used in calculating the transport rate). A section is included in Chapter 5 summarizing the classical Reference 3 and 4 gravity model used in most inertial navigation systems. A navigation coordinate frame version of the gravity model is developed in Chapter 5 for input to the Chapter 4 velocity rate equation (in a form known as "plumb-bob gravity" which includes a centripetal acceleration term associated with position relative to the rotating earth). In all cases, the equations are developed in complete closed-form without resorting to "first order approximations" prevalent in many navigation analytical documents. It is the author's belief that computer technology has now advanced sufficiently (in speed, word length, and higher order language utilization) that closed-form equations can be implemented without penalty in an INS computer. Significant advantages thereby accrue in algorithm performance and in accompanying software validation/documentation processes that need not address the accuracy of first order approximations. Chapter 5 concludes with a summary table listing the Chapter 5 equations that would be utilized in a typical INS computer.

In a strapdown INS there are three operations that must be initialized prior to engaging the inertial navigation "mode". These are the integration functions used to determine attitude, velocity and position. Chapter 6 addresses the analytics associated with performing the attitude/velocity/position integration function initialization operations in the INS computer for applications when the user vehicle is at a "quasi-stationary" attitude/position orientation (e.g., an aircraft on the ground with parking brake engaged, but under quasi-stationary attitude/position motion due to wind gusts, passenger/crew movement, fuel/stores loading). The attitude initialization process discussed utilizes a closed-loop "Kalman filter" aided integration process using inputs from the INS accelerometers and angular rate sensors to initialize the attitude orientation relative to the vertical and true north ("true heading"). The true heading initialization is achieved by estimating horizontal earth rotation rate components and using the result to initialize the heading attitude of the INS attitude parameters or the orientation of the navigation coordinate frame (i.e., the position direction cosine matrix). Chapter 6 also analytically describes a "Coarse Leveling" process by which an approximate vertical attitude initialization can be achieved using accelerometer inputs. Coarse Leveling is typically performed before engaging the previously described vertical/heading initialization process (known as "Fine Alignment").

Chapter 7 derives the equivalent digital integration algorithm form of the Chapter 4 differential equations for attitude/velocity/position determination in the strapdown INS computer. The attitude algorithm development section addresses both direction cosine matrix and quaternion forms for strapdown sensor attitude relative to the locally level navigation frame, each separately dealing with updating for strapdown sensor rotation (measured by the angular rate sensors) and for navigation frame rotation rates. The attitude algorithms are structured based on three repetition rates (per pass of the associated computation chain); a high speed rate for high frequency angular rate sensor inputs (e.g., angular vibration), a moderate speed attitude updating rate for angular rate sensor inputs (including summing of the high speed algorithm output), and a lower speed attitude updating rate for navigation frame rotation rates. Closed-form expressions (without approximation) are derived for all but the high speed algorithms. The high speed algorithm is derived as an approximation to an exact continuous form integral equation that measures what is known as "coning" effects in the attitude solution.

The velocity updating algorithms in Chapter 7 are also structured using a multiple speed architecture; a high rate algorithm to measure high frequency effects and a moderate speed algorithm to handle the velocity updating operation (including summing of the high speed algorithm output). Closed-form expressions (without approximation) are derived for all but the high speed algorithm. The high speed algorithm uses angular rate sensor and accelerometer inputs in an approximation of an exact continuous integral equation to measure what is known as "sculling" effects in the acceleration-transformation/velocity-updating operation. The moderate speed algorithm adds the sculling output from the high speed algorithm to summed increments of integrated accelerometer specific force output (including what is known as a "rotation compensation" correction), transforms the result to the navigation frame, adds plumb-bob gravity, and adds Coriolis acceleration effects (to account for navigation and earth rotation rate effects) to update the navigation frame velocity components.

Two forms of position updating algorithms are presented in Chapter 7; a classical set operated at a single repetition rate based on trapezoidal integration of velocity, and a "high resolution" set based on a multiple speed architecture similar to the attitude/velocity updating algorithms. For the high resolution approach, a high rate algorithm measures high frequency effects and a moderate speed algorithm handles the position updating operation (including summing of the high speed algorithm output). Closed-form expressions (without approximation) are derived for all but the high speed algorithm. The high speed algorithm uses angular rate sensor, accelerometer, and sculling algorithm inputs to measure what has been termed (by the author) "scrolling effects" in the position updating process. The lower speed algorithm adds the scrolling output from the high speed algorithm to computed increments of doubly integrated accelerometer specific force output (including a "position rotation compensation" correction - author coined name), transforms the result to the navigation frame, adds the position change due to velocity at the start of the update cycle, and uses the resulting navigation frame "position change increment" to update the position data (altitude and the position direction cosine matrix). The trapezoidal positioning algorithm computations are identical to the moderate speed portion of the high resolution algorithms, but with the position change increment calculated as a trapezoidal integration approximation for integrated navigation frame velocity.

For the Chapter 7 attitude, velocity and position multiple speed algorithms, the form of the algorithms is structured so that in situations when sufficient throughput exists (the trend for the future), the lower speed algorithms can be executed at the higher speed algorithm repetition rate to simplify the software executive control architecture. A table is provided at the conclusion of Chapter 7 listing the Chapter 7 and Chapter 5 computation algorithms that would be typically used in a high performance INS, in their order of execution in the INS computer.

A fundamental problem with all inertial navigation systems is the inability to manufacture inertial components with the inherent accuracy required to meet system requirements. To correct for this deficiency, compensation algorithms are included in the INS software for correcting the sensor outputs for known predictable error effects. Chapter 8 develops analytical equations for compensating the strapdown inertial sensor outputs. Inertial sensor compensation algorithms derive from classical analytical models used in the inertial sensor industry to characterize a sensor's output (including errors) as a function of the sensor input (error free). In contrast, the sensor compensation algorithms used in the INS computer are designed to translate the sensor outputs (containing error) to the equivalent error free form. Thus, the compensation algorithms represent the inverse of the inertial sensor analytical model equations. In many systems, the form of the compensation equations so derived contain linearization approximations to the exact inverse relations (to conserve on computer throughput). The approach taken in Chapter 8 is to use the complete inverse form (without approximation) based on the assumption that modern day computers of today (and certainly in the future) can handle the workload.

Chapter 8 is divided into four parts; development of the inertial sensor output compensation algorithms, developing algorithms for correcting the high speed portion of the attitude/velocity/position algorithms (i.e., coning, sculling, scrolling, integration of inertial sensor outputs between computation cycles) that may have been calculated using uncompensated inertial sensor data, compensation for misalignment of the strapdown sensor assembly relative to the INS mount installation in the user vehicle, and a summary section. The summary provides a tabulated listing of the compensation equations that might be used in a high performance INS, tabulated in the order of execution in the INS computer, and showing their application in conjunction with the Chapter 5 and 7 inertial navigation computation algorithms. Chapter 8 includes a discussion of methods for compensating quantization error on the strapdown inertial sensor signals. Also included is the derivation of algorithms for compensating the effect of physical displacement between the accelerometers in a strapdown sensor assembly (known as "size effect") which, under angular rotation, exposes each accelerometer to a slightly different acceleration vector. Intermediate computation results in the size effect algorithms are also applied for compensating anisoinertia angular rate sensitive error effects in pendulous accelerometers.

In some applications (e.g., Synthetic Aperture Radar), it is important that jitter motion of the strapdown inertial sensor assembly be removed from the computed INS attitude/velocity/position outputs. Chapter 9 provides a smoothing architecture for achieving such jitter compensation that avoids introducing dynamic distortion in the smoothed output signals.

Chapter 10 develops analytical techniques for evaluating the error in the high speed portion of the attitude/velocity/position algorithms under anticipated sinusoidal and random INS input vibrations. Strapdown inertial navigation integration algorithms are designed to accurately account for three-dimensional high frequency angular and linear vibration of the sensor assembly. Such motion, if not properly accounted for, can lead to systematic attitude/velocity/position error build-up. The high speed algorithms developed in Chapter 7 to measure these effects (i.e., coning, sculling and high resolution position updating routines) are based on approximations to the form of the angular-rate/specific-force profiles during the high speed update interval. An important part of the algorithm design is their accuracy evaluation under hypothesized vibration exposures of the strapdown INS in the user vehicle, the subject of

Chapter 10. Algorithm performance evaluation results, used in design/synthesis iterative fashion, eventually set the order of the algorithm selected and its required repetition rate in the INS computer.

Since the sensor assembly is dynamically coupled to the INS mount through the INS structure (in many cases including mechanical isolators and their imbalances), vibrations input to the INS mount become dynamically distorted as they translate into the resulting inertial sensor output vibrations provided to the navigation algorithms. Included in Chapter 10 is a review of linear dynamic system frequency response analytics and the development of a simplified analytical model for characterizing the dynamic response of an INS sensor assembly to input vibration. The sensor assembly dynamic response model is one of the elements utilized in the Chapter 10 algorithm performance evaluation equations presented. Chapter 10 includes an analysis of folding effect amplification in the position update algorithms induced by linear vibrations of the sensor assembly. Such effects are generally not present in the attitude/velocity algorithms because the inertial sensors are generally of the integrating type, providing their inputs to the navigation computer in the form of pre-integrated angular rate and acceleration increments. Chapter 10 also provides an analysis of coning/sculling algorithm error induced by inertial sensor dynamic mismatch. Chapter 10 concludes with an analytical description of a simple simulation program that can be used to evaluate high speed algorithm error under user specified INS sinusoidal and random vibration input exposure.

Chapter 11 deals with the validation of strapdown inertial navigation integration algorithms by computer simulation. It addresses the basic issue of how to analytically generate a "truth model" set of angular rate and specific force acceleration data representative of the output from ideal (error free) strapdown inertial sensors (typically in the form of integrated angular rate and specific force acceleration increments), and how to analytically generate a corresponding "truth model" attitude/velocity/position profile. Validation of the algorithms then consists of running the algorithms at their selected repetition rate(s) using the "truth model" sensor inputs, and comparing the algorithm attitude/velocity/position response to the equivalent "truth model" attitude/velocity/position profile. In general, two methods can be considered for the truth model; 1. A digital integration approach in which the truth model integration algorithms are more accurate than the INS algorithms being validated, and 2. Closed-form analytical equations representing the exact analytical integration of the angular-rate/specific-force profile. The problem with the Method 1 approach is the dilemma it presents in demonstrating the accuracy of a truth model that also contains digital integration algorithm error, <u>allegedly</u> smaller than the error in the INS digital integration algorithms being tested. Chapter 11 addresses the Method 2 approach, and derives closed-form analytically exact truth models for evaluating classical groupings of INS algorithms used to execute basic integration operations; 1. Attitude updating under dynamic conditions, 2. Attitude updating, acceleration transformation, velocity updating under dynamic conditions, 3. Attitude updating, acceleration transformation, velocity/position updating under dynamic conditions (including accelerometer size effect separation), 4. Attitude/velocity/position updating during long term navigation over an ellipsoidal earth model. Simulation programs for these functions are analytically described in Chapter 11 including the method of comparing the INS algorithm results with the truth model. A table is provided showing which algorithm equations (by Equation number used in the book) are validated with each simulator. Chapter 11 also includes a discussion of specialized simulators for use in validating particular algorithm portions.

The overall strapdown INS design process requires supporting analyses to develop and verify performance specifications for the INS components, particularly the inertial sensors. This generally entails the use of a strapdown INS error model in the form of time rate differential equations that describe the error response in INS computed attitude/velocity/position data. Such error models are also fundamental to the design of Kalman filters (the subject of Chapter 15) used, in conjunction with other system inputs, for correcting the INS errors. Chapter 12 provides a detailed description of the analytical process used in deriving strapdown INS error model equations that represent the INS integration routine response to sensor input errors (i.e., excluding the effect of algorithm and computer finite word-length error, errors that are generally negligible in a well designed modern day INS compared to sensor error effects). Chapter 12 is based on the error form of the Chapter 4 and 5 strapdown INS computation equations.

An important part of INS error model development is the definition (and selection) of attitude/velocity/position error parameters used in the error model and their relationship to the INS computed attitude/velocity/position parameters (or to a hypothetical set of INS navigation parameters that are analytically related to the INS computed set). Chapter 12 describes several versions of navigation error parameters that can be considered and the process followed in selecting one set for a particular application. After describing the general process used in developing INS error models, Chapter 12 derives error model equations for different error parameter sets using two approaches; 1. Direct derivation based on the error parameter definitions, 2. Derivation by conversion of a previously derived error model (based on one set of error parameters) to an alternative error model based on another error parameter set. The second method is based on equivalencies between error parameter sets derived in Chapter 12. Included in Chapter 12 is the analytical modeling of inertial sensor error inputs and modeling of error effects induced by sensor assembly vibration.

Chapter 13 deals with analytical solutions to the Chapter 12 strapdown INS error model equations under classical trajectory profiles and inertial sensor error characteristics. Such analyses are useful for understanding the nature of sensor error propagation into attitude/velocity/position under particular conditions, and for pencil-and-paper performance predictions. Chapter 13 begins with a general analytical description of INS error characteristics including vertical channel response (with gravity model induced exponential divergence/control), horizontal channel response (Schuler oscillations and long term "earth loop" effects), and the unique characteristics of strapdown inertial sensor scale-factor/misalignment error on INS navigation performance under dynamic angular rate conditions. Chapter 13 then develops closed-form solutions to the Chapter 12 equations under various simplifying assumptions for the sensor errors and trajectory profile (e.g., constant sensor errors under high rate spinning about stationary and rotating axes, in horizontal circular trajectories (in general and at Schuler frequency), and under long term cruise; random sensor error effects during short term and 2 hour trajectories).

Chapter 14 addresses the effect of strapdown inertial sensor error on the Chapter 6 quasi-stationary initial vertical/heading attitude initialization process. The error model for the Chapter 6 Fine Alignment initialization process equations is developed and solved in closed-form for constant and random output inertial sensor errors, ramping accelerometer error, inertial sensor quantization errors, and random vibrations. The random and quantization error analysis is based on solving the general continuous form Kalman filter covariance differential equation developed in Chapter 15.

The accuracy of all inertial navigation systems is fundamentally limited by instabilities in the inertial component error characteristics following calibration. Resulting residual inertial sensor errors produce INS navigation errors that are unacceptable in many applications. To overcome these deficiencies, "inertial aiding" is commonly utilized in which the INS navigation parameters (and in some cases, the sensor calibration coefficients) are updated based on measurements from an alternate source of navigation information available in the user vehicle (e.g., Global Position System (GPS) receiver provided data). The modern method for applying the inertial aiding measurement to the INS data is through a Kalman filter, a set of software that is typically resident in the INS computer. Chapter 15 describes Kalman filtering in general and how it relates to the aiding of strapdown inertial navigation systems. Included is a detailed introductory section that develops the basic theory of Kalman filter estimation in general, its interface/timing/synchronization architecture in the host computer, and procedures for software validation.

The Kalman filter theory developed in Chapter 15 is at the on-set, based on "optimally" estimating an "error state vector" representing the error characteristics of the device(s) providing inputs. This contrasts with classical Kalman filter theory based on estimating a "state vector" representing parameters in the input devices (e.g., position parameters in an INS). In inertial navigation applications, the error state vector selected for the Kalman filter is related to the computed navigation parameters (e.g., the three component attitude error vector described in Chapter 12 which is related to errors in the nine component attitude direction matrix or the four component attitude quaternion), but generally does not explicitly represent the errors in the computed parameters (as in the traditional "extended" Kalman filter approach). Developing the Kalman filter from scratch based on a general error state vector approach provides a direct method for arriving at the result used in most Kalman filter applications.

Chapter 15 develops discrete recursive forms of the Kalman filter (suitable for software implementation) and a general continuous form for performance analysis. Following the approach outlined in Reference 6, a general solution is developed for the continuous form Kalman filter. The result is then extended for the singular case of zero "measurement noise", a situation encountered in Kalman filters applied to the Chapter 6 inertial navigation system Fine Alignment process (and used in Chapter 14 to derive closed-form solutions for the Chapter 6 quasi-stationary initial alignment error equations). Chapter 15 provides examples of discrete form Kalman filter configurations applied to the Chapter 6 Fine Alignment process, to dynamic moving base INS initial alignment, to inertial aiding using a generic vehicle mounted velocity sensor measurement, and for inertial aiding using GPS position range measurements.

Inherent in the structure of Kalman filters is a statistical estimate of the uncertainties in the computed error state vector, typically represented in the form of an error state uncertainty "covariance matrix". The calculations involved in continuously computing the covariance matrix (it changes as a function of time based on the time profile of the navigation parameters, system/sensor error characteristics, and Kalman updating history) can also be used in a performance analysis time domain simulation program for statistically estimating the INS errors. Chapter 16 addresses the structure of such covariance simulation programs for application to INS performance assessment and as part of the Kalman filter design process. As a Kalman filter design aid, the covariance simulation is used to simulate the equivalent operations performed in the Kalman filter being designed/tested and, from a covariance standpoint, to evaluate the performance of the INS when aided by the Kalman filter. The aided INS performance analysis capability permits the user to account for all error effects being simulated (the so-called "real world" model) when interfaced with a Kalman filter based on an approximate version of the real world (the so-called "suboptimal" Kalman filter). The Kalman filter design process consists of using the simulation over a representative set of trajectory profiles, evaluating aided INS performance, and modifying the Kalman filter dynamic model (e.g., the number of error state vector elements or the magnitudes of included noise sources) in iterative fashion until performance satisfies user specified criteria. Included in the Chapter 16 covariance simulation program, is the ability to provide sensitivity outputs that identify the sensitivity of navigation errors to the error sources, and an "error budget" that shows the contribution of each error source to the navigation errors, both of which are useful during the Kalman filter design/iteration process.

Simulation analysis of strapdown inertial navigation systems often require the use of "trajectory generators", simulation programs that provide navigation parameter outputs as a function of time over a user selected trajectory profile. The Chapter 16 covariance simulation program requires such a trajectory generator input as does the process described in Chapter 15 for validating Kalman filters (and their internal computational elements). Chapter 17 deals with the design of trajectory generators that provide navigation parameter outputs as well as strapdown inertial sensor inputs in the form of integrated angular-rate/specific-force-acceleration increments (integrals between trajectory generator time points). The integrated inertial sensor increments are identical to the outputs from idealized strapdown inertial sensors (i.e., error free), with the trajectory generator navigation parameters then representing the output from an idealized error free strapdown inertial navigation system. Chapter 17 first describes the general requirements for a trajectory: 1. Trajectory shaping, an interactive process by which the user creates a trajectory profile to meet a general set of requirements, and 2. Trajectory regeneration in which the shaped trajectory is "played back" as part of a larger simulation program to regenerate the navigation/inertial sensor data history. Included must be the inherent characteristic of the navigation parameter outputs (attitude/velocity/position) and associated inertial sensor signals to be consistent with what would be obtained from an ideal integration of the inertial sensor data into trajectory navigation parameters.

The major portion of Chapter 17 provides a detailed description of a trajectory generator designed to produce realistic trajectories representative of maneuvering vehicles in the vicinity of the earth (i.e., aircraft, surface ships, underwater vehicles). Once the trajectory profile is created, the Chapter 17 trajectory generator provides (as options in the trajectory regeneration process) the ability to add aerodynamic angle-of-attack/sideslip effects, user vehicle structural bending effects, high frequency vibrations, and to simulate trajectories of different points in the same vehicle separated by flexible structure. The Chapter 17 trajectory regeneration function is structured as the analytical inverse of the Chapter 7 high accuracy strapdown inertial navigation algorithms (including high resolution position updating). This technique assures that integration of the trajectory generator inertial sensor signals with the Chapter 7 algorithms will produce the same navigation solution, the correct response under error free sensor and computer processing conditions.

Chapter 18 describes five system level tests that can be performed on a strapdown INS to ascertain the error characteristics of the strapdown inertial sensors; the Schuler Pump Test, Strapdown Drift Test, Repeated Alignment Test, Continuous Alignment Test and the Strapdown Rotation Test. Each can be executed in a test laboratory using a rotation fixture to which the INS is mounted. The Schuler Pump Test is based on amplifying the classic 84 minute sinusoidal Schuler error response characteristic of a strapdown INS (described in Chapter 12). Analysis of the velocity error response provides the ability to determine composite angular rate sensor and accelerometer errors that created it. The Strapdown Drift test is a static test in which the attitude integration software in the INS computer is configured to constrain the average horizontal transformed specific force acceleration to zero. For a test of several hours duration, the averages of the constraining signals become accurate measures of angular rate sensor bias error. The Repeated Alignment Test is a static test in which the Chapter 6 Fine Alignment process is repeated to generate a sample set of horizontal earth rate estimates at the end of alignment. By analyzing the variance in the end-of-alignment earth rate signals, the horizontal angular rate sensor random noise is estimated. The Continuous Alignment Test estimates horizontal angular rate sensor random noise using the time history of horizontal earth rate estimates taken during a single initial alignment run. The Strapdown Rotation Test consists of exposing the INS to a series of rotations, and recording its average transformed specific force acceleration output at static dwell times between rotations. By processing the recorded data, very accurate measurements can be made of the scale factor error and relative misalignment for all inertial sensors in the sensor assembly, the accelerometer bias errors, and misalignment of the sensor assembly relative to the INS mounting fixture. In each case, the test procedure is described and the analytics developed in detail for the associated data processing algorithms.

Chapter 19 provides three pertinent papers published by the author since the original publication of this book in 2000. The first paper derives from velocity/position algorithms developed in Chapter 7 that are designed to be exact under particular trajectory conditions (primarily, constant strapdown angular rate and specific force over the velocity update interval). Using the exact velocity/position updating algorithm structure as a base, high speed routines are derived for computing the algorithm input under general trajectory conditions. The result is a two speed velocity/position algorithm structure that directly parallels the two-speed attitude updating approach described in Chapter 7. The second paper provides an integrated and expanded treatment of material on sensor quantization error described in several sections of this book. Of particular interest are new sections rigorously describing how quantization error is properly modeled to account for different attitude/velocity/position algorithm update rates. The third paper addresses some fundamental questions on implicit assumptions used throughout the book regarding inertial sensor measurements. Gyros measure angular rate relative to non-rotating inertial space. Accelerometers measure specific force which when analytically combined with gravitational acceleration provide total acceleration for integration into velocity/position. Specific force has been defined as the acceleration relative to non-rotating inertial space produced by non-gravitational forces. But what exactly is non-rotating inertial space? What exactly is total acceleration? Is gravitation an absolute or a relative parameter? Can specific force be defined without reference to gravity? The third paper in Chapter 19 provides some interesting answers to these and other fundamental inertial sensing questions.

With the exception of this Chapter 1 Introduction, each chapter includes an introductory Overview section outlining the basic material to be covered. References for all chapters are provided in the back of each of the Part 1 and 2 book volumes.

2 Terminology

2.0 OVERVIEW

Due to the diversity and scope of analytical topics treated in this book, coupled with traditional analytical symbol usage in various areas and the limited number of symbols available for use, the quest for a common symbol terminology throughout the book has been virtually impossible to achieve. Unfortunately, symbols in some chapters of the book had to be used in other chapters with different meanings. In general, however, the symbol definitions in particular sections and chapters have single meanings that are generally consistent with common usage (as appropriate). Two basic techniques have been incorporated to provide clarity in symbol definitions throughout the book and to avoid the common need to search previous sections and pages for the definition of symbols in current use: 1. Providing liberal definitions of symbols in each section following equations in which they first appear (including repeats of important previously provided definitions that are far removed from the current text), and 2. Providing an alphabetized index for all symbols defined in the book, identifying the immediately preceding equation, figure or section number (as appropriate) where their definitions are provided.

This chapter provides an overview of the various analytical symbols used in the book. Section 2.1 provides a general definition of mathematical notation utilized. A tabular listing is included in Section 2.1 containing an alphabetized listing of all mathematical notations defined throughout the book, identifying where their definitions are provided. Section 2.2 defines the coordinate frames most commonly used throughout the book for describing vector components and attitude orientations. Section 2.2 also describes the Coordinate Frame Index provided in the back of the book containing an alphabetized listing of all coordinate frames defined throughout the book, identifying where their definitions are provided. Section 2.3 describes the Parameter Index provided in the back of the book containing an alphabetized listing of all analytical parameters defined throughout the book (exclusive of coordinate frames), identifying where their definitions are provided. The section, figure and equation numbering convention selected for the book provides the key for finding the location of analytical definitions identified in the Section 2.1 - 2.3 alphabetical tabular listings.

Section numbers in the book begin with the chapter number in which the section appears (e.g., Section 7.1.2.4 is found in Chapter 7 as the fourth subsection of Section 7.1.2, Section 7.1.2 is the second subsection under Section 7.1 and Section 7.1 is the first technical subsection in Chapter 7, following the Section 7.0 Overview). The .0 section designation is reserved for the overview that appears at the beginning of each chapter (with the exception of Chapter 1 which has no overview).

Figure and table numbers in the book are numbered consecutively beginning with 1 for each section. Each figure (or table) number begins with the section number where it is presented, followed by the figure (or table) number (e.g., Figure 3.2.2-1 is the first numbered figure found in Section 3.2.2; Table 2.1-1 is the first numbered table in Section 2.1).

Equation numbers in the book are numbered consecutively beginning with 1 for each section, and are also enclosed in brackets (e.g., Equation (7.1.2.3-8) is the eighth numbered equation in Section 7.1.2.3).

Definitions for analytical terms provided throughout the book are preceded by the key word "where" that immediately follows the equation, figure, or section number in which the analytical term appears. For example, immediately following Equations (6.1.2-2) in Section 6.1.2 we find the phrase:

"where

$\underline{u}_{ZN}^{N}$ = Unit vector along the N Frame vertical axis (Z), projected on N Frame axes.

K_1, K_2, K_3, K_4 = Fine Alignment process estimation feedback control gains.

Etc."

The $\underline{u}_{ZN}^{N}$, K_1, K_2, K_3, K_4, etc. terms appear in Equations (6.1.2-2).

The above described numbering convention allows a simple method for locating the definition for a particular analytical term. One first locates the term in one of the three appropriate alphabetical tabular listings described in Sections 2.1 - 2.3 to follow. The tabular listing identifies the equation, figure, or section number immediately preceding the definition of the analytical term. One then finds the identified equation, figure, or section number in the text, finds the "where" word immediately following the number, and then locates the definition for the analytical term of interest following the "where" word.

2.1 MATHEMATICAL NOTATION

A thorough introduction to the origin, meaning and use of the basic notation utilized throughout the book is presented in Chapter 3. The principal mathematical notation is similar to that first introduced in Reference 3, and is as follows:

Coordinate Frame = Analytical abstraction defined by three consecutively numbered (or lettered) unit vectors that are mutually perpendicular to one another in the right hand sense. A coordinate frame can be visualized as a set of three perpendicular lines (axes) passing through a common point (origin) with the unit vectors emanating from the origin along the axes. In this book, the physical position of each coordinate frame's origin is arbitrary.

$\underline{V}$ = Vector without specific coordinate frame designation. A vector is a parameter that has length and direction. The vectors used in this book are classified as "free vectors", hence, have no preferred location in coordinate frames in which they are analytically described.

$\underline{V}^A$ = Column matrix with elements equal to the projection of $\underline{V}$ on Coordinate Frame A axes. The projection of $\underline{V}$ on each Frame A axis equals the dot product of $\underline{V}$ with the coordinate Frame A axis unit vector. The elements of $\underline{V}^A$ are also identified as the components of $\underline{V}$ in coordinate Frame A.

$\left(\underline{V}^A \times\right)$ = Skew symmetric (or cross-product) form of $\underline{V}^A$ represented by the square matrix $\begin{bmatrix} 0 & -V_{ZA} & V_{YA} \\ V_{ZA} & 0 & -V_{XA} \\ -V_{YA} & V_{XA} & 0 \end{bmatrix}$ in which V_{XA}, V_{YA}, V_{ZA} are the components of $\underline{V}^A$. The matrix product of $\left(\underline{V}^A \times\right)$ with another A Frame vector column matrix equals the cross-product of $\underline{V}$ with the vector in the A Frame; i.e., $\left(\underline{V}^A \times\right) \underline{W}^A = \left(\underline{V} \times \underline{W}\right)^A$.

$C_{A_2}^{A_1}$ = Direction cosine matrix that transforms a vector from its Coordinate Frame A_2 projection form to its Coordinate Frame A_1 projection form.

$\underline{\omega}_{A_1 A_2}$ = Angular rate of Coordinate Frame A_2 relative to Coordinate Frame A_1. When A_1 is a non-rotating inertial coordinate frame, $\underline{\omega}_{A_1 A_2}$ is the angular rate that would be measured by angular rate sensors mounted on Frame A_2.

$\underline{a}_{SF}$ = Specific force acceleration defined as the acceleration relative to non-rotating inertial space produced by applied non-gravitational forces. Accelerometers measure $\underline{a}_{SF}$.

t = Time.

$\dot{( )} = \frac{d( )}{dt}$ = Derivative with respect to time.

$\equiv$ Equal by definition.

$|\;|$ = Absolute value of a scalar quantity or the magnitude ("length") of a vector.

Table 2.1-1 provides a complete listing of the mathematical symbols defined throughout the book, identifying all book locations where their definitions are provided (by preceding equation or section number).

Table 2.1-1 Mathematical Symbols

Symbol	Preceding Equation, Figure, Or Section No.
$\|\;\|$	Sect. No. 2.1
$d(\;)$	(5.2.4-7)
∇	(5.1-2)
$\dfrac{\partial(\;)}{\partial \underline{P}}$	(17.1.2.3-32)
$\dfrac{\partial(\;)}{\partial[\;]}$	(17.1.2.3-32)
$\Delta(\;)$	(16.2.5-1)
$\ddot{(\;)}$	(4.4.1.2.1-5)
$\dddot{(\;)}$	(4.4.1.2.1-6)
$\dot{(\;)}$	Sect. No. 2.1
$\equiv$	Sect. No. 2.1
$\overset{o}{=}$	(8.1.4.1.2-3)
$\widehat{\;}$	(3.5.2-1)
$\widehat{\;}$	(7.1.1.3-1)
$\widehat{\;}$	(7.1.2.3-1)
$\widehat{\;}$	(12.2.1-4)
$\widehat{\;}$	(12.3.1-2)
$\widehat{\;}$	(15.2.2.1-7)
$\widehat{\;}$	(18.2.1-1)
$\widehat{\;}$	(18.3-1)
$\widehat{\;}$	(18.3.1.2-1)
$\widehat{\;}$	(18.4.7-2)
$\widehat{\;}$	Sect. No. 3.5
(-)	(15.1-4)
(-)	(15.1.2-13)
(-)	(15.1.2.1.1.4-3)
(-)	(15.1.2.3-3)
(-)	(15.1.5.3.2-6)
(-)	(16.1.1-5)
-	(7.1.1.3-3)
-	(7.1.2.3-8)
-	(7.2-6)
-	(7.3.1-5)
(+)	(15.1.2.1.1.4-3)
+	(7.1.1.3-3)
+	(7.1.2.3-8)
+	(7.2-6)
+	(7.3.1-5)
$(+_c)$	(15.1-4)
$(+_c)$	(15.1.2-13)
$(+_c)$	(15.1.2.3-3)
$(+_c)$	(15.1.5.3.2-1)
$(+_c)$	(16.1.1-5)
$(+_e)$	(15.1.2-13)
$(+_e)$	(15.1.2.3-1)
$(+_e)$	(15.1.5.3.2-6)
$(+_e)$	(16.1.1-12)

´	(8.1.2.1-6)
´	(8.1.4.1.4-3)
′	(14.6.1-8)
'	(16.2.3.3-2)
"	(16.1.1.4-4)
Sign ( )	(3.2.2.2-19)
Sign ( )	(8.1.1.3-11)
Sign ( )	(8.1.3.2-2)
Sign ( )	(10.6.2-9)
Sign ( )	(15.1.2.1.1.4-3)
Sign ( )	(18.4.7.1-3)
Sign ($\underline{V}$)	(8.1.3.3-6)
*	(15.2.1.1-11)
*	(16.1.1-12)
*	(16.2.3-6)
Step( )	(8.1.3.2-1)
Step( )	(18.1.1-3)
T	(3.1-12)
T	(3.5.1-2)
~	(12.3.1-2)
~	(12.4-5)
~	(13.4.1.1-2)
~	(15.1.2-13)
~	(15.2.2.1-26)
~	(18.2.1-1)
~	(18.3-1)
$(\underline{V}^A \cdot\ )$	(3.1.1-11)
$(\underline{V}^A \times)$	(3.1.1-13)

2.2 COORDINATE FRAME DEFINITIONS

A complete alphabetized listing of the coordinate frames defined throughout the book is provided in the Coordinate Frame Index at the end of each of the book's Part 1 and Part 2 volumes (following the Subject Index), identifying all book locations where their definitions are provided (by preceding equation, figure or section number).

The following defines the most commonly used coordinate frames:

E Frame = Earth fixed coordinate frame used for position location definition. Typically defined with one axis parallel to the earth polar axis with the other axes fixed to the earth and parallel to the equatorial plane.

N Frame = Navigation coordinate frame having its Z axis parallel to the upward vertical at the local earth surface referenced position location point on the earth's surface. Used for integrating acceleration into velocity and for defining the angular orientation of the local vertical in the E Frame.

L Frame = Locally level coordinate frame parallel to the N Frame but with Z axis parallel to the downward vertical, and X, Y axes along N Frame Y, X axes. Used as the reference for describing the strapdown sensor coordinate frame orientation.

Geo Frame = Locally level geographic coordinate frame defined with its Z axis upward along the local geodetic vertical, Y axis north (and horizontal) with X axis east (and horizontal).

B Frame = Strapdown inertial sensor coordinates ("body frame") with axes parallel to nominal right handed orthogonal sensor input axes. This definition applies to Chapters 4 - 18. In Chapter 3, the B Frame is an arbitrary coordinate frame used to describe generalized vector/matrix characteristics.

I Frame = Non-rotating inertial coordinate frame used as a reference for angular rotation measurements. Particular orientations selected for the I Frame are discussed in the sections when their orientation is pertinent to analytical operations.

2.3 PARAMETER DEFINITIONS

The Parameter Index included at the end of each of the book's Part 1 and Part 2 volumes (following the Subject and Coordinate Frame Indexes) provides a complete alphabetized listing of the computational parameters defined throughout the book (exclusive of coordinate frames), identifying all book locations where their definitions are provided (by preceding equation, figure or section number).

The convention followed in the alphabetization process is to order numerical digits first (in numerical order with 0 first) and Arabic letters second (in alphabetical order) with upper case letters preceding lower case letters. There is no discrimination between underlined versus non-underlined identical characters, or between identical characters of different Font size or physical position (subscripted, superscripted or normal). When identical characters appear in several book locations, they are listed in the Parameter Index in order of Equation number first, Figure number second and Section number last. Grammatical, mathematical and other special symbols appearing in a particular parameter are treated uniquely in the alphabetization process for each type. They should not cause difficulty in locating a particular parameter in the Parameter Index if they are ignored; the parameter of interest with its special symbols will be found nearby and can easily be identified by a quick visual search.

Complex parameters with superscripts and subscripts are alphabetized considering the main character(s) first, the subscripted character(s) second and the superscripted character(s) last. For example, the character $\underline{a}_{SF_{XForm}}^{A}$ would be alphabetized according to the name $\underline{a}$SFXFormA.

The convention for alphabetizing Greek letters in the Parameter Index is according to its complete Arabic spelling as summarized in Table 2.3-1.

Table 2.3-1 Greek Symbols Utilized

GREEK SYMBOL		ARABIC SPELLING	
Upper Case	**Lower Case**	**Upper Case**	**Lower Case**
Not Used	α	Alpha	alpha
Not Used	β	Beta	beta
Γ	γ	Gamma	gamma
Δ	δ	Delta	delta
Not Used	ε	Epsilon	epsilon
Not Used	ζ	Zeta	zeta
Not Used	η	Eta	eta
ϑ or Θ	θ	Theta	theta
Not Used	Not Used	Iota	iota
Not Used	κ	Kappa	kappa
Λ	λ	Lambda	lambda
Not Used	μ	Mu	mu
Not Used	ν	Nu	nu
Not Used	ξ	Xi	xi
Not Used	Not Used	Omicron	omicron
Π	π	Pi	pi
Not Used	ρ	Rho	rho
Σ	σ	Sigma	sigma
Not Used	τ	Tau	tau
Not Used	υ	Upsilon	upsilon
Φ	ϕ or φ	Phi	phi
Not Used	χ	Chi	chi
Ψ	ψ	Psi	psi
Ω	ω	Omega	omega

Thus, according to Table 2.3-1 and the alphabetizing convention described previously, the parameter $\Delta\underline{\alpha}^{L}_{\omega EB_H}$ would be alphabetized according to the name DeltaalphaomegaEBHL

The $\perp$ and ∂ symbols in the Parameter Index are alphabetized according to “Perp” (for $\perp$) and “del” (for ∂).

3 Vector, Attitude And Coordinate Frame Fundamentals

3.0 OVERVIEW

Strapdown inertial navigation system analysis deals with the description of vector and attitude orientation parameters, their rates of change and associated error characteristics. In this chapter we provide a detailed introduction to the origin, meaning and use of the basic analytical notation applied throughout the book for strapdown system analysis.

Section 3.1 describes the basic concept of a vector, its mathematical description in selected coordinate frames, and its transformation properties between coordinate frames. Section 3.2 provides an analytical description of four commonly used methods for describing the relative attitude orientation between two coordinate frames: the direction cosine matrix, the rotation vector, Euler angles, and the attitude quaternion. Included are analytical methods for converting one attitude form to another, and methods for transforming vectors between coordinate frames using direction cosines or the attitude quaternion.

Section 3.3 derives equations that describe the rate of change of the Section 3.2 attitude orientation parameters. Section 3.4 develops equations for describing the rate of change of vector components in one coordinate frame in terms of its component rates of change in another frame when there is an angular rotation rate between the two coordinate frames.

Section 3.5 provides a detailed analytical discussion of the error characteristics associated with vector and attitude orientation parameters, with particular emphasis on fundamental definitions for error parameters described in various coordinate frames and their associated transformation properties between coordinate frames.

3.1 VECTORS AND COORDINATE FRAME TRANSFORMATIONS

Consider an arbitrary vector expressed as the sum of its components projected on the axes of an arbitrary coordinate frame:

$$\underline{V} = V_{XA}\, \underline{u}_{XA} + V_{YA}\, \underline{u}_{YA} + V_{ZA}\, \underline{u}_{ZA} \tag{3.1-1}$$

where

$\underline{V}$ = Arbitrary vector.

A = Arbitrary right handed coordinate frame.

$\underline{u}_{XA}$, $\underline{u}_{YA}$, $\underline{u}_{ZA}$ = Unit vectors along coordinate Frame A axes X, Y, and Z.

V_{XA}, V_{YA}, V_{ZA} = Projections of $\underline{V}$ on Frame A axes X, Y, and Z defined as the magnitude (or length) of $\underline{V}$ multiplied by the cosine of the angles between $\underline{V}$ and X, Y or Z.

The same vector $\underline{V}$ can also be expressed in terms of its projections along the axes of another arbitrary coordinate frame:

$$\underline{V} = V_{XB}\,\underline{u}_{XB} + V_{YB}\,\underline{u}_{YB} + V_{ZB}\,\underline{u}_{ZB} \tag{3.1-2}$$

where

B = Another arbitrary right handed orthogonal coordinate frame.

$\underline{u}_{XB}$, $\underline{u}_{YB}$, $\underline{u}_{ZB}$ = Unit vectors along Frame B axes X, Y, and Z.

V_{XB}, V_{YB}, V_{ZB} = Projections of $\underline{V}$ on Frame B axes X, Y, and Z.

Classical vector algebra (Reference 38, Section 13-3) defines the dot product between two arbitrary vectors $\underline{V}$ and $\underline{W}$ to be:

$$\underline{V} \cdot \underline{W} = V\,W \cos\phi \tag{3.1-3}$$

where

V, W = Magnitude (or length) of $\underline{V}$, $\underline{W}$.

ϕ = Angle between $\underline{V}$ and $\underline{W}$.

An expression for V_{XA} in terms of the $\underline{V}$ components in Frame B is obtained by equating Equations (3.1-1) and (3.1-2) and taking the dot product of the result with $\underline{u}_{XA}$. Since the magnitude of $\underline{u}_{XA}$ is one, we then have with (3.1-3):

$$V_{XA} = V_{XB}\,\underline{u}_{XA} \cdot \underline{u}_{XB} + V_{YB}\,\underline{u}_{XA} \cdot \underline{u}_{YB} + V_{ZB}\,\underline{u}_{XA} \cdot \underline{u}_{ZB} \tag{3.1-4}$$

Performing similar operations for the Y and Z components yields the following for the Frame A components of $\underline{V}$ in terms of the Frame B components:

$$\begin{aligned}
V_{XA} &= V_{XB}\,\underline{u}_{XA} \cdot \underline{u}_{XB} + V_{YB}\,\underline{u}_{XA} \cdot \underline{u}_{YB} + V_{ZB}\,\underline{u}_{XA} \cdot \underline{u}_{ZB} \\
V_{YA} &= V_{XB}\,\underline{u}_{YA} \cdot \underline{u}_{XB} + V_{YB}\,\underline{u}_{YA} \cdot \underline{u}_{YB} + V_{ZB}\,\underline{u}_{YA} \cdot \underline{u}_{ZB} \\
V_{ZA} &= V_{XB}\,\underline{u}_{ZA} \cdot \underline{u}_{XB} + V_{YB}\,\underline{u}_{ZA} \cdot \underline{u}_{YB} + V_{ZB}\,\underline{u}_{ZA} \cdot \underline{u}_{ZB}
\end{aligned} \tag{3.1-5}$$

The Frame B components as a function of the Frame A components are obtained similarly using dot products with the Frame B unit vectors:

$$
\begin{aligned}
V_{XB} &= V_{XA}\, \underline{u}_{XB} \cdot \underline{u}_{XA} + V_{YA}\, \underline{u}_{XB} \cdot \underline{u}_{YA} + V_{ZA}\, \underline{u}_{XB} \cdot \underline{u}_{ZA} \\
V_{YB} &= V_{XA}\, \underline{u}_{YB} \cdot \underline{u}_{XA} + V_{YA}\, \underline{u}_{YB} \cdot \underline{u}_{YA} + V_{ZA}\, \underline{u}_{YB} \cdot \underline{u}_{ZA} \\
V_{ZB} &= V_{XA}\, \underline{u}_{ZB} \cdot \underline{u}_{XA} + V_{YA}\, \underline{u}_{ZB} \cdot \underline{u}_{YA} + V_{ZA}\, \underline{u}_{ZB} \cdot \underline{u}_{ZA}
\end{aligned}
\tag{3.1-6}
$$

The dot product between a coordinate axis unit vector in Frame A and a Frame B coordinate axis unit vector equals the cosine of the angle between the respective Frame A and Frame B axes, and is identified as the direction cosine between the respective axes. Equations (3.1-5) and (3.1-6) can be compressed if we then adopt the following notation for the A and B Frame unit vector dot products:

$$
C_{IJ} \equiv \underline{u}_{IA} \cdot \underline{u}_{JB} = \underline{u}_{JB} \cdot \underline{u}_{IA}
\tag{3.1-7}
$$

where

C_{IJ} = Direction cosine between axis I of Frame A and axis J of Frame B.

With the above definition, Equations (3.1-5) and (3.1-6) become:

$$
\begin{aligned}
V_{XA} &= C_{11}\, V_{XB} + C_{12}\, V_{YB} + C_{13}\, V_{ZB} \\
V_{YA} &= C_{21}\, V_{XB} + C_{22}\, V_{YB} + C_{23}\, V_{ZB} \\
V_{ZA} &= C_{31}\, V_{XB} + C_{32}\, V_{YB} + C_{33}\, V_{ZB}
\end{aligned}
\tag{3.1-8}
$$

$$
\begin{aligned}
V_{XB} &= C_{11}\, V_{XA} + C_{21}\, V_{YA} + C_{31}\, V_{ZA} \\
V_{YB} &= C_{12}\, V_{XA} + C_{22}\, V_{YA} + C_{32}\, V_{ZA} \\
V_{ZB} &= C_{13}\, V_{XA} + C_{23}\, V_{YA} + C_{33}\, V_{ZA}
\end{aligned}
\tag{3.1-9}
$$

Equations (3.1-8) and (3.1-9) are now in a form that can be adapted to standard matrix algebra notation (Reference 37, Chapter 4 - Section 16). We first define the following matrices:

$$
\underline{V}^A \equiv \begin{bmatrix} V_{XA} \\ V_{YA} \\ V_{ZA} \end{bmatrix} \qquad \underline{V}^B \equiv \begin{bmatrix} V_{XB} \\ V_{YB} \\ V_{ZB} \end{bmatrix} \qquad C_B^A \equiv \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}
\tag{3.1-10}
$$

where

C_B^A = Direction cosine matrix that converts vector components from the Frame B to the Frame A projection form.

Substituting the Equation (3.1-10) definitions In (3.1-8) and (3.1-9) then yields the following equivalent matrix expressions:

$$\underline{V}^A = C_B^A \, \underline{V}^B \qquad \underline{V}^B = C_A^B \, \underline{V}^A \tag{3.1-11}$$

in which, from Equations (3.1-8) - (3.1-10),

$$C_A^B = \left(C_B^A\right)^T \qquad C_B^A = \left(C_A^B\right)^T \tag{3.1-12}$$

where

T = Superscript designating matrix transpose.

Equations (3.1-11) are known as "vector transformation" operations. Based on Equations (3.1-11), we can also refine our definition of a vector expressed as a column matrix, and a direction cosine matrix for transforming the vector, where:

$\underline{V}^A$, $\underline{V}^B$ = Column matrix vectors whose elements equal the projections of vector $\underline{V}$ on coordinate Frame A and B axes.

C_B^A = Direction cosine matrix that will transform a column matrix vector from coordinate Frame B to coordinate Frame A.

Let us now substitute the Equations (3.1-11) $\underline{V}^B$ expression into the $\underline{V}^A$ equation, using (3.1-12) for C_A^B. The result is:

$$\underline{V}^A = C_B^A \, \underline{V}^B = C_B^A \, C_A^B \, \underline{V}^A = C_B^A \left(C_B^A\right)^T \underline{V}^A \tag{3.1-13}$$

From Equation (3.1-13) it is clear that:

$$C_B^A \left(C_B^A\right)^T = I \tag{3.1-14}$$

where

I = Identity matrix.

Multiplying (3.1-14) by the inverse of C_B^A then obtains:

$$\left(C_B^A\right)^{-1} = \left(C_B^A\right)^T \tag{3.1-15}$$

Similar operations applied to the $\underline{V}^B$ expression in Equations (3.1-11) yield the equivalent relation for the C_A^B matrix:

$$\left(C_A^B\right)^{-1} = \left(C_A^B\right)^T \tag{3.1-16}$$

Equations (3.1-15) and (3.1-16) show that the inverse of a direction cosine matrix equals its transpose.

Let us now transform one of the Frame B coordinate axis unit vectors into Frame A and substitute the expression for C_B^A from (3.1-10):

$$\underline{u}_{JB}^A = C_B^A \, \underline{u}_{JB}^B = \begin{bmatrix} C_{1J} \\ C_{2J} \\ C_{3J} \end{bmatrix} \tag{3.1-17}$$

where

J = Index of 1, 2 or 3 corresponding to B Frame X, Y or Z axes.

Hence, for the C_B^A definition in (3.1-10):

$$C_B^A = \left[\underline{u}_{1B}^A \quad \underline{u}_{2B}^A \quad \underline{u}_{3B}^A \right] \tag{3.1-18}$$

Equation (3.1-18) shows that the columns of C_B^A represent coordinate axis unit vectors in Frame B projected on Frame A axes.

Similarly, the converse operation for the Frame A coordinate axis unit vectors using C_A^B from Equations (3.1-12) yields:

$$\underline{u}_{IA}^B = C_A^B \, \underline{u}_{IA}^A = \left(C_B^A\right)^T \underline{u}_{IA}^A = \begin{bmatrix} C_{I1} \\ C_{I2} \\ C_{I3} \end{bmatrix} \tag{3.1-19}$$

and

$$C_A^B = \left[\underline{u}_{1A}^B \quad \underline{u}_{2A}^B \quad \underline{u}_{3A}^B \right] \tag{3.1-20}$$

where

I = Index of 1, 2 or 3 corresponding to A Frame axis X, Y or Z.

Equation (3.1-20) shows that the columns of C_A^B represent Frame A coordinate axis unit vectors projected on Frame B axes.

Now consider yet another arbitrary coordinate frame for which we write the equivalent to (3.1-11) for $\underline{V}^B$:

$$\underline{V}^B = C_D^B \, \underline{V}^D \tag{3.1-21}$$

where

D = Yet another arbitrary right handed coordinate frame.

C_D^B = Direction cosine matrix that transforms vectors from the D to the B Frame.

Substituting Equation (3.1-21) into the Equations (3.1-11) $\underline{V}^A$ expression yields:

$$\underline{V}^A = C_B^A \, C_D^B \, \underline{V}^D \tag{3.1-22}$$

But,

$$\underline{V}^A = C_D^A \, \underline{V}^D \tag{3.1-23}$$

where

C_D^A = Direction cosine matrix that transforms vectors from the D to the A Frame.

Equating (3.1-22) and (3.1-23) shows that:

$$C_D^A = C_B^A \, C_D^B \tag{3.1-24}$$

which is the chain rule for direction cosine matrix products.

3.1.1 VECTOR PRODUCT OPERATORS AND TRANSFORMATION CHARACTERISTICS

Consider two arbitrary vectors with components defined in an arbitrary coordinate Frame A:

$$\begin{aligned} \underline{V} &= V_{XA} \, \underline{u}_{XA} + V_{YA} \, \underline{u}_{YA} + V_{ZA} \, \underline{u}_{ZA} \\ \underline{W} &= W_{XA} \, \underline{u}_{XA} + W_{YA} \, \underline{u}_{YA} + W_{ZA} \, \underline{u}_{ZA} \end{aligned} \tag{3.1.1-1}$$

where

$\underline{V}, \underline{W}$ = Arbitrary vectors.

$\underline{u}_{XA}, \underline{u}_{YA}, \underline{u}_{ZA}$ = Unit vectors along Frame A coordinate axes.

V_{XA}, V_{YA}, V_{ZA} = Components of $\underline{V}$ in Frame A.

W_{XA}, W_{YA}, W_{ZA} = Components of $\underline{W}$ in Frame A.

Repeating Equation (3.1-3), the definition of the dot product between $\underline{V}$ and $\underline{W}$ is:

$$\underline{V} \cdot \underline{W} = V\,W \cos\phi \tag{3.1.1-2}$$

Note from its definition, that the dot product is a scalar quantity.

The definition of the cross-product (Reference 38, Section 13-4) between $\underline{V}$ and $\underline{W}$ is:

$$\underline{V} \times \underline{W} = \underline{n}\, V\,W \sin\phi \tag{3.1.1-3}$$

where

$\underline{n}$ = Unit vector perpendicular to $\underline{V}$ and $\underline{W}$ whose positive direction is defined by the right hand rule that curls the fingers of the right hand from $\underline{V}$ into $\underline{W}$, with positive $\underline{n}$ thereby provided in the thumb pointing direction.

Note from its definition, that the cross product is a vector quantity. Also note that Equations (3.1.1-2) and (3.1.1-3), the definition for $\underline{n}$, and the definition for ϕ (following Equation (3.1-3)) do no specify whether the angle selected for ϕ (i.e., between $\underline{V}$ and $\underline{W}$) is greater or less than π (i.e., there are two ways to measure the angle between $\underline{V}$ and $\underline{W}$; the "short" angular way when ϕ is less than π, and the "long" angular way in which ϕ is $2\,\pi$ minus the shorter ϕ). Equations (3.1.1-2) and (3.1.1-3) are compatible with either definition because $\cos\phi$ in (3.1.1-2) equals $\cos(2\,\pi - \phi)$, $\sin\phi$ in (3.1.1-3) equals $-\sin(2\,\pi - \phi)$, and the direction of $\underline{n}$ in (3.1.1-3) (by the previous $\underline{n}$ definition) for the "shorter" angular direction "from $\underline{V}$ into $\underline{W}$", is the negative of the $\underline{n}$ vector corresponding to the "longer" angular direction "from $\underline{V}$ into $\underline{W}$". We suffer no loss of generality if we consider ϕ to be less than π which then defines the $\underline{n}$ vector from the right hand rule using the shorter angular distance "from $\underline{V}$ into $\underline{W}$". By this selection, $\sin\phi$ in (3.1.1-3) is always positive.

Classical vector algebra and geometry (Reference 38, Sections 13-2 through 13-4) shows that:

$$V = \sqrt{V_{XA}^2 + V_{YA}^2 + V_{ZA}^2} \qquad W = \sqrt{W_{XA}^2 + W_{YA}^2 + W_{ZA}^2} \tag{3.1.1-4}$$

$$\underline{V} \cdot \underline{W} = V_{XA}\, W_{XA} + V_{YA}\, W_{YA} + V_{ZA}\, W_{ZA} \tag{3.1.1-5}$$

$$\begin{aligned} \underline{V} \times \underline{W} = & \left(V_{YA}\; W_{ZA} - V_{ZA}\; W_{YA}\right) \underline{u}_{XA} \\ & + \left(V_{ZA}\; W_{XA} - V_{XA}\; W_{ZA}\right) \underline{u}_{YA} + \left(V_{XA}\; W_{YA} - V_{YA}\; W_{XA}\right) \underline{u}_{ZA} \end{aligned} \tag{3.1.1-6}$$

We also note from Equations (3.1.1-5) and (3.1.1-6) that:

$$\underline{V} \cdot \underline{W} = \underline{W} \cdot \underline{V} \tag{3.1.1-7}$$

$$\underline{V} \times \underline{W} = - \underline{W} \times \underline{V} \tag{3.1.1-8}$$

As in Section 3.1, we now utilize a matrix format to define $\underline{V}$ and $\underline{W}$ in Frame A:

$$\underline{V}^A \equiv \begin{bmatrix} V_{XA} \\ V_{YA} \\ V_{ZA} \end{bmatrix} \qquad \underline{W}^A \equiv \begin{bmatrix} W_{XA} \\ W_{YA} \\ W_{ZA} \end{bmatrix} \tag{3.1.1-9}$$

Using the basic rules of matrix algebra (Reference 37, Chapter 4 - Section 16) we see that the equivalent to dot product Equation (3.1.1-5) is:

$$\underline{V}^A \cdot \; \underline{W}^A \equiv \left(\underline{V}^A\right)^T \underline{W}^A \tag{3.1.1-10}$$

or, if we define a dot product operator, Equation (3.1.1-10) is equivalently:

$$\left(\underline{V}^A \cdot\right) \underline{W}^A \equiv \underline{V}^A \cdot \underline{W}^A \tag{3.1.1-11}$$

where

$\left(\underline{V}^A \cdot\right) =$ Dot product operator associated with $\underline{V}^A$.

Comparing Equations (3.1.1-11) and (3.1.1-10) we see that:

$$\left(\underline{V}^A \cdot\right) \equiv \left(\underline{V}^A\right)^T \tag{3.1.1-12}$$

We also can define a cross-product operator for $\underline{V}^A$ such that:

$$\left(\underline{V}^A \times\right) \underline{W}^A \equiv \underline{V}^A \times \underline{W}^A \tag{3.1.1-13}$$

where

$\left(\underline{V}^A \times\right)$ = Cross product operator associated with $\underline{V}^A$.

Applying the standard rules for matrix products (Reference 37, Chapter 4 - Section 16), we see from Equations (3.1.1-6) and (3.1.1-13) that:

$$\left(\underline{V}^A \times\right) \equiv \begin{bmatrix} 0 & -V_{ZA} & V_{YA} \\ V_{ZA} & 0 & -V_{XA} \\ -V_{YA} & V_{XA} & 0 \end{bmatrix} \tag{3.1.1-14}$$

From Equation (3.1.1-14) we see that $\left(\underline{V}^A \times\right)$ is a skew symmetric matrix; the diagonal elements are zero and the upper right off-diagonal elements equal the negative of the lower left off-diagonal elements. For this reason, the $\left(\underline{V}^A \times\right)$ matrix is sometimes referred to as the skew symmetric form of $\underline{V}^A$.

A useful property of the product of skew symmetric and standard vector forms is the following (as can be verified by individual component substitution and expansion):

$$\underline{V}_1^A \times \left(\underline{V}_2^A \times \underline{V}_3^A\right) = \left(\underline{V}_1^A \times\right)\left[\left(\underline{V}_2^A \times\right)\underline{V}_3^A\right] = \left[\left(\underline{V}_1^A \times\right)\left(\underline{V}_2^A \times\right)\right]\underline{V}_3^A = \left(\underline{V}_1^A \times\right)\left(\underline{V}_2^A \times\right)\underline{V}_3^A \tag{3.1.1-15}$$

where

$\underline{V}_1^A, \underline{V}_2^A, \underline{V}_3^A$ = Arbitrary vectors projected on arbitrary Frame A axes.

It will also be useful to develop an expression for the cross-product operator associated with the cross-product of two vectors. This is accomplished using from general vector algebra (Reference 37, Chapter 4 - Section 6), the vector triple cross product identity:

$$\underline{V}_1 \times \left(\underline{V}_2 \times \underline{V}_3\right) = \underline{V}_2\left(\underline{V}_1 \cdot \underline{V}_3\right) - \underline{V}_3\left(\underline{V}_1 \cdot \underline{V}_2\right) \tag{3.1.1-16}$$

where

$\underline{V}_1, \underline{V}_2, \underline{V}_3$ = Arbitrary vectors.

By interchanging indices 1 and 2, Equation (3.1.1-16) is equivalently:

$$\underline{V}_2 \times \left(\underline{V}_1 \times \underline{V}_3\right) = \underline{V}_1\left(\underline{V}_2 \cdot \underline{V}_3\right) - \underline{V}_3\left(\underline{V}_2 \cdot \underline{V}_1\right) \tag{3.1.1-17}$$

By exchanging indices 3 for 1, 1 for 2 and 2 for 3, and applying Equations (3.1.1-7) and (3.1.1-8), Equation (3.1.1-16) can also be written as:

$$\left(\underline{V}_1 \times \underline{V}_2\right) \times \underline{V}_3 \; = \; - \, \underline{V}_1 \left(\underline{V}_2 \cdot \underline{V}_3\right) + \underline{V}_2 \left(\underline{V}_1 \cdot \underline{V}_3\right) \qquad (3.1.1\text{-}18)$$

We now take the difference between Equations (3.1.1-16) and (3.1.1-17) to obtain, with (3.1.1-18):

$$\begin{aligned} \underline{V}_1 \times \left(\underline{V}_2 \times \underline{V}_3\right) - \underline{V}_2 \times \left(\underline{V}_1 \times \underline{V}_3\right) \; &= \; \underline{V}_2 \left(\underline{V}_1 \cdot \underline{V}_3\right) - \underline{V}_1 \left(\underline{V}_2 \cdot \underline{V}_3\right) \\ &= \; \left(\underline{V}_1 \times \underline{V}_2\right) \times \underline{V}_3 \end{aligned} \qquad (3.1.1\text{-}19)$$

With (3.1.1-13), Equation (3.1.1-19) becomes after rearrangement in our arbitrary A Frame:

$$\left[\left(\underline{V}_1^A \times \underline{V}_2^A\right) \times\right] \underline{V}_3^A \; = \; \left(\underline{V}_1^A \times\right) \left[\left(\underline{V}_2^A \times\right) \underline{V}_3^A\right] - \left(\underline{V}_2^A \times\right) \left[\left(\underline{V}_1^A \times\right) \underline{V}_3^A\right] \qquad (3.1.1\text{-}20)$$

or with (3.1.1-15):

$$\left[\left(\underline{V}_1^A \times \underline{V}_2^A\right) \times\right] \underline{V}_3^A \; = \; \left(\underline{V}_1^A \times\right) \left(\underline{V}_2^A \times\right) \underline{V}_3^A - \left(\underline{V}_2^A \times\right) \left(\underline{V}_1^A \times\right) \underline{V}_3^A \qquad (3.1.1\text{-}21)$$

Since $\underline{V}_3^A$ was defined as an arbitrary vector, Equation (3.1.1-21) simplifies to the following expression for the skew symmetric form of the cross-product between two vectors:

$$\left[\left(\underline{V}_1^A \times \underline{V}_2^A\right) \times\right] \; = \; \left(\underline{V}_1^A \times\right) \left(\underline{V}_2^A \times\right) - \left(\underline{V}_2^A \times\right) \left(\underline{V}_1^A \times\right) \qquad (3.1.1\text{-}22)$$

To develop the coordinate frame transformation characteristics of $\left(\underline{V}^A \cdot\right)$ and $\left(\underline{V}^A \times\right)$, we reintroduce Equations (3.1-11) from Section 3.1 applied to $\underline{V}$ and $\underline{W}$:

$$\underline{V}^A \; = \; C_B^A \, \underline{V}^B \qquad \qquad \underline{V}^B \; = \; C_A^B \, \underline{V}^A \qquad (3.1.1\text{-}23)$$

$$\underline{W}^A \; = \; C_B^A \, \underline{W}^B \qquad \qquad \underline{W}^B \; = \; C_A^B \, \underline{W}^A \qquad (3.1.1\text{-}24)$$

where

C_B^A, C_A^B = Direction cosine matrices that transform vectors from arbitrary coordinate Frame B to arbitrary coordinate Frame A, and from Frame A to Frame B.

Applying (3.1.1-23) to (3.1.1-12) shows that:

$$\left(\underline{V}^A \cdot\right) \; = \; \left(\underline{V}^A\right)^T \; = \; \left[\left(C_B^A \, \underline{V}^B\right) \cdot\right] \; = \; \left(C_B^A \, \underline{V}^B\right)^T \; = \; \left(\underline{V}^B\right)^T \left(C_B^A\right)^T \qquad (3.1.1\text{-}25)$$

or, with Equations (3.1-12) and (3.1.1-12):

$$\left(\underline{V}^A \cdot\right) = \left(\underline{V}^B \cdot\right) C_A^B \tag{3.1.1-26}$$

Similarly:

$$\left(\underline{V}^B \cdot\right) = \left(\underline{V}^A \cdot\right) C_B^A \tag{3.1.1-27}$$

Equations (3.1.1-26) and (3.1.1-27) are the dot product operator transformation equivalents to Equations (3.1.1-23).

Applying Equations (3.1.1-23), (3.1.1-24) and (3.1-15) to Equation (3.1.1-10) also shows that:

$$\begin{aligned} \underline{V}^A \cdot \underline{W}^A &= \left(\underline{V}^A\right)^T \underline{W}^A = \left(C_B^A\, \underline{V}^B\right)^T C_B^A\, \underline{W}^B \\ &= \left(\underline{V}^B\right)^T \left(C_B^A\right)^T C_B^A\, \underline{W}^B = \left(\underline{V}^B\right)^T \underline{W}^B \end{aligned} \tag{3.1.1-28}$$

or, with (3.1.1-10):

$$\underline{V}^A \cdot \underline{W}^A = \underline{V}^B \cdot \underline{W}^B \tag{3.1.1-29}$$

Hence, the dot product is identical in all coordinate frames ("invariant").

In order to investigate the transformation characteristics of $\left(\underline{V}^A \times\right)$, we reintroduce Equation (3.1-20) with (3.1-12) from Section 3.1:

$$\left(C_B^A\right)^T = \begin{bmatrix} \underline{u}_{1A}^B & \underline{u}_{2A}^B & \underline{u}_{3A}^B \end{bmatrix} \qquad C_B^A = \begin{bmatrix} \left(\underline{u}_{1A}^B\right)^T \\ \left(\underline{u}_{2A}^B\right)^T \\ \left(\underline{u}_{3A}^B\right)^T \end{bmatrix} \tag{3.1.1-30}$$

where

$\underline{u}_{1A}^B, \underline{u}_{2A}^B, \underline{u}_{3A}^B$ = Unit vectors along A Frames axes 1, 2, 3 (i.e., X, Y, Z) projected on Frame B axes.

Substituting (3.1.1-30) for C_B^A in (3.1.1-23) and applying (3.1.1-10) obtains:

$$\underline{V}^A = C_B^A\, \underline{V}^B = \begin{bmatrix} \left(\underline{u}_{1A}^B\right)^T \underline{V}^B \\ \left(\underline{u}_{2A}^B\right)^T \underline{V}^B \\ \left(\underline{u}_{3A}^B\right)^T \underline{V}^B \end{bmatrix} = \begin{bmatrix} \underline{u}_{1A}^B \cdot \underline{V}^B \\ \underline{u}_{2A}^B \cdot \underline{V}^B \\ \underline{u}_{3A}^B \cdot \underline{V}^B \end{bmatrix} \tag{3.1.1-31}$$

With (3.1.1-31), the $\left(\underline{V}^A \times\right)$ cross-product operator becomes from (3.1.1-14):

$$\left(\underline{V}^A \times\right) = \left[\left(C_B^A \, \underline{V}^B\right) \times\right] = \begin{bmatrix} 0 & -\,\underline{u}_{3A}^B \cdot \underline{V}^B & \underline{u}_{2A}^B \cdot \underline{V}^B \\ \underline{u}_{3A}^B \cdot \underline{V}^B & 0 & -\,\underline{u}_{1A}^B \cdot \underline{V}^B \\ -\,\underline{u}_{2A}^B \cdot \underline{V}^B & \underline{u}_{1A}^B \cdot \underline{V}^B & 0 \end{bmatrix} \tag{3.1.1-32}$$

From the definitions of $\underline{u}_{1A}^B, \underline{u}_{2A}^B, \underline{u}_{3A}^B$, we can write:

$$\begin{aligned} \underline{u}_{1A}^B &= \underline{u}_{2A}^B \times \underline{u}_{3A}^B = -\,\underline{u}_{3A}^B \times \underline{u}_{2A}^B \\ \underline{u}_{2A}^B &= \underline{u}_{3A}^B \times \underline{u}_{1A}^B = -\,\underline{u}_{1A}^B \times \underline{u}_{3A}^B \\ \underline{u}_{3A}^B &= \underline{u}_{1A}^B \times \underline{u}_{2A}^B = -\,\underline{u}_{2A}^B \times \underline{u}_{1A}^B \\ \underline{u}_{1A}^B \times \underline{u}_{1A}^B &= \underline{u}_{2A}^B \times \underline{u}_{2A}^B = \underline{u}_{3A}^B \times \underline{u}_{3A}^B = 0 \end{aligned} \tag{3.1.1-33}$$

With (3.1.1-33), Equation (3.1.1-32) can be expanded to the following equivalent form:

$$\left[\left(C_B^A \, \underline{V}^B\right) \times\right] = \begin{bmatrix} \left(\underline{u}_{1A}^B \times \underline{u}_{1A}^B\right) \cdot \underline{V}^B & \left(\underline{u}_{2A}^B \times \underline{u}_{1A}^B\right) \cdot \underline{V}^B & \left(\underline{u}_{3A}^B \times \underline{u}_{1A}^B\right) \cdot \underline{V}^B \\ \left(\underline{u}_{1A}^B \times \underline{u}_{2A}^B\right) \cdot \underline{V}^B & \left(\underline{u}_{2A}^B \times \underline{u}_{2A}^B\right) \cdot \underline{V}^B & \left(\underline{u}_{3A}^B \times \underline{u}_{2A}^B\right) \cdot \underline{V}^B \\ \left(\underline{u}_{1A}^B \times \underline{u}_{3A}^B\right) \cdot \underline{V}^B & \left(\underline{u}_{2A}^B \times \underline{u}_{3A}^B\right) \cdot \underline{V}^B & \left(\underline{u}_{3A}^B \times \underline{u}_{3A}^B\right) \cdot \underline{V}^B \end{bmatrix} \tag{3.1.1-34}$$

From general vector algebra (Reference 37, Chapter 4 - Section 6), the mixed vector dot/cross product identity states that:

$$\left(\underline{V}_1 \times \underline{V}_2\right) \cdot \underline{V}_3 = \left(\underline{V}_2 \times \underline{V}_3\right) \cdot \underline{V}_1 = \left(\underline{V}_3 \times \underline{V}_1\right) \cdot \underline{V}_2 \tag{3.1.1-35}$$

where

$\underline{V}_1, \underline{V}_2, \underline{V}_3$ = Arbitrary vectors.

Applying (3.1.1-35) to (3.1.1-34) then obtains:

$$\left[\left(C_B^A \underline{V}^B\right)\times\right] = \begin{bmatrix} \left(\underline{u}_{1A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{1A}^B\right) & \left(\underline{u}_{1A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{2A}^B\right) & \left(\underline{u}_{1A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{3A}^B\right) \\ \left(\underline{u}_{2A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{1A}^B\right) & \left(\underline{u}_{2A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{2A}^B\right) & \left(\underline{u}_{2A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{3A}^B\right) \\ \left(\underline{u}_{3A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{1A}^B\right) & \left(\underline{u}_{3A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{2A}^B\right) & \left(\underline{u}_{3A}^B\right)\cdot\left(\underline{V}^B\times\underline{u}_{3A}^B\right) \end{bmatrix} \qquad (3.1.1\text{-}36)$$

or, with (3.1.1-12) and (3.1.1-13):

$$\left[\left(C_B^A\, \underline{V}^B\right)\times\right] = \begin{bmatrix} \left(\underline{u}_{1A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{1A}^B & \left(\underline{u}_{1A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{2A}^B & \left(\underline{u}_{1A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{3A}^B \\ \left(\underline{u}_{2A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{1A}^B & \left(\underline{u}_{2A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{2A}^B & \left(\underline{u}_{2A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{3A}^B \\ \left(\underline{u}_{3A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{1A}^B & \left(\underline{u}_{3A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{2A}^B & \left(\underline{u}_{3A}^B\right)^T\left(\underline{V}^B\times\right)\underline{u}_{3A}^B \end{bmatrix} \qquad (3.1.1\text{-}37)$$

If we utilize Equations (3.1.1-30) for C_B^A and $\left(C_B^A\right)^T$ we would find that the matrix product group $C_B^A\left(\underline{V}^B\times\right)\left(C_B^A\right)^T$ is exactly equal to the right side of Equation (3.1.1-37). We thereby conclude from (3.1.1-37) that:

$$\left[\left(C_B^A\, \underline{V}^B\right)\times\right] = C_B^A\left(\underline{V}^B\times\right)\left(C_B^A\right)^T \qquad (3.1.1\text{-}38)$$

or with (3.1.1-23):

$$\left(\underline{V}^A\times\right) = C_B^A\left(\underline{V}^B\times\right)\left(C_B^A\right)^T \qquad (3.1.1\text{-}39)$$

Similar analysis also reveals that:

$$\left(\underline{V}^B\times\right) = C_A^B\left(\underline{V}^A\times\right)\left(C_A^B\right)^T \qquad (3.1.1\text{-}40)$$

Equations (3.1.1-39) and (3.1.1-40) are the cross product operator transformation equivalents to Equations (3.1.1-23). The general matrix product form on the right of Equations (3.1.1-39) and (3.1.1-40) is denoted as a similarity transformation used to transform a dyadic from one coordinate frame to another. In fact, the formal definition of a dyadic is a parameter that transforms between coordinate frames with a similarity transformation. Equations (3.1.1-39) and (3.1.1-40) show that a similarity transformation can be utilized to transform a cross-product operator from one coordinate frame to its equivalent form in another. We conclude, therefore, that the cross-product operator is a dyadic.

Equation (3.1.1-39) applied to another arbitrary vector $\underline{W}$ provides an equivalent vector cross-product relationship. Multiplying (3.1.1-39) by $\underline{W}^A$ and applying (3.1.1-24) yields:

$$\left(\underline{V}^{A}\times\right)\underline{W}^{A} \;=\; C_{B}^{A}\left(\underline{V}^{B}\times\right)\left(C_{B}^{A}\right)^{T}\underline{W}^{A} \;=\; C_{B}^{A}\left(\underline{V}^{B}\times\right)\underline{W}^{B} \tag{3.1.1-41}$$

or with (3.1.1-13):

$$\underline{V}^{A}\times\underline{W}^{A} \;=\; C_{B}^{A}\left(\underline{V}^{B}\times\underline{W}^{B}\right) \tag{3.1.1-42}$$

Thus, the transform of the cross-product between two vectors equals the cross-product of the transformed vectors.

Another useful transformation relationship involves the component of an arbitrary vector $\underline{V}$ that is perpendicular to some arbitrarily defined unit vector $\underline{u}$. The component of $\underline{V}$ perpendicular to $\underline{u}$ equals $\underline{V}$ minus the component of $\underline{V}$ along $\underline{u}$, hence, in our arbitrary A and B Frames:

$$\underline{V}_{\perp}^{A} \;=\; \underline{V}^{A} - \left(\underline{V}^{A}\cdot\underline{u}^{A}\right)\underline{u}^{A} \tag{3.1.1-43}$$

$$\underline{V}_{\perp}^{B} \;=\; \underline{V}^{B} - \left(\underline{V}^{B}\cdot\underline{u}^{B}\right)\underline{u}^{B} \tag{3.1.1-44}$$

where

$\underline{V}_{\perp}$ = Component of $\underline{V}$ perpendicular to $\underline{u}$.

Let's now transform (3.1.1-44) into the A Frame (using (3.1.1-23) and (3.1.1-29)) to obtain:

$$\begin{aligned} C_{B}^{A}\,\underline{V}_{\perp}^{B} \;&=\; C_{B}^{A}\left[\underline{V}^{B} - \left(\underline{V}^{B}\cdot\underline{u}^{B}\right)\underline{u}^{B}\right] \;=\; \underline{V}^{A} - \left(\underline{V}^{B}\cdot\underline{u}^{B}\right)\underline{u}^{A} \\ &=\; \underline{V}^{A} - \left(\underline{V}^{A}\cdot\underline{u}^{A}\right)\underline{u}^{A} \;=\; \underline{V}_{\perp}^{A} \end{aligned} \tag{3.1.1-45}$$

But we also know from (3.1.1-23) that:

$$\underline{V}_{\perp}^{A} \;=\; \left(C_{B}^{A}\,\underline{V}^{B}\right)_{\perp} \tag{3.1.1-46}$$

Combining (3.1.1-45) and (3.1.1-46) then shows that:

$$\left(C_{B}^{A}\,\underline{V}^{B}\right)_{\perp} = C_{B}^{A}\,\underline{V}_{\perp}^{B} \tag{3.1.1-47}$$

Thus, the same result is obtained if we first find the $\underline{V}$ component perpendicular to $\underline{u}$ and then transform the perpendicular component to another coordinate frame, or if we first transform $\underline{V}$ and then find its component perpendicular to $\underline{u}$ in the new frame.

3.2 ATTITUDE PARAMETERS

Four commonly used parameters for representing the relative angular orientation between two coordinate frames are direction cosines (i.e., the elements of the direction cosine matrix), the rotation vector, Euler angles and the attitude quaternion. Analytical properties of each are discussed in the following subsections as well as analytical equivalencies between them.

3.2.1 DIRECTION COSINES

The direction cosine matrix is frequently applied in strapdown inertial system analysis to describe the relative attitude between two coordinate frames. It has the advantage of being easily used to transform vectors between the two frames. From Sections 3.1 and 3.1.1, the properties of the direction cosine matrix are as follows:

$$\underline{V}^A \equiv \begin{bmatrix} V_{XA} \\ V_{YA} \\ V_{ZA} \end{bmatrix} \qquad \underline{V}^B \equiv \begin{bmatrix} V_{XB} \\ V_{YB} \\ V_{ZB} \end{bmatrix}$$

$$\left(\underline{V}^A \cdot\right) \equiv \left(\underline{V}^A\right)^T \qquad \left(\underline{V}^B \cdot\right) \equiv \left(\underline{V}^B\right)^T \tag{3.2.1-1}$$

$$\left(\underline{V}^A \times\right) \equiv \begin{bmatrix} 0 & -V_{ZA} & V_{YA} \\ V_{ZA} & 0 & -V_{XA} \\ -V_{YA} & V_{XA} & 0 \end{bmatrix} \qquad \left(\underline{V}^B \times\right) \equiv \begin{bmatrix} 0 & -V_{ZB} & V_{YB} \\ V_{ZB} & 0 & -V_{XB} \\ -V_{YB} & V_{XB} & 0 \end{bmatrix}$$

$$C_B^A \equiv \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}$$

$$\underline{V}^A = C_B^A \, \underline{V}^B \qquad \underline{V}^B = C_A^B \, \underline{V}^A \tag{3.2.1-2}$$

$$C_A^B = \left(C_B^A\right)^T \qquad C_B^A = \left(C_A^B\right)^T \tag{3.2.1-3}$$

$$\left(C_B^A\right)^{-1} = \left(C_B^A\right)^T \qquad \left(C_A^B\right)^{-1} = \left(C_A^B\right)^T \tag{3.2.1-4}$$

$$C_D^A = C_B^A \, C_D^B \tag{3.2.1-5}$$

$$C_B^A = \left[\underline{u}_{1B}^A \;\; \underline{u}_{2B}^A \;\; \underline{u}_{3B}^A \right] \qquad C_A^B = \left[\underline{u}_{1A}^B \;\; \underline{u}_{2A}^B \;\; \underline{u}_{3A}^B \right] \tag{3.2.1-6}$$

$$\left(\underline{V}^A \cdot\right) = \left(\underline{V}^B \cdot\right) C_A^B \qquad \left(\underline{V}^B \cdot\right) = \left(\underline{V}^A \cdot\right) C_B^A \tag{3.2.1-7}$$

$$\left(\underline{V}^A \times\right) = C_B^A \left(\underline{V}^B \times\right) \left(C_B^A\right)^T \qquad \left(\underline{V}^B \times\right) = C_A^B \left(\underline{V}^A \times\right) \left(C_A^B\right)^T \tag{3.2.1-8}$$

where

$\underline{V}^A, \underline{V}^B$ = Column matrices whose elements equal the projections of vector $\underline{V}$ on coordinate Frame A and B axes.

$C_B^A, C_A^B, C_D^A, C_D^B$ = Direction cosine matrices that transform vectors from coordinate Frame B to A, from coordinate Frame A to B, from coordinate Frame D to A, and from coordinate Frame D to B.

V_{XA}, V_{YA}, V_{ZA} = Projections of $\underline{V}$ on Frame A axes X, Y, and Z.

V_{XB}, V_{YB}, V_{ZB} = Projections of $\underline{V}$ on Frame B axes X, Y, and Z.

C_{IJ} = Direction cosine between axis I of Frame A and axis J of Frame B.

$\underline{u}_{1B}^A, \underline{u}_{2B}^A, \underline{u}_{3B}^A$ = Unit vectors along Frame B axes X, Y, Z (i.e., 1, 2, 3) as projected on Frame A (superscript) coordinate axes.

$\underline{u}_{1A}^B, \underline{u}_{2A}^B, \underline{u}_{3A}^B$ = Unit vectors along Frame A axes X, Y, Z (i.e., 1, 2, 3) as projected on Frame B (superscript) coordinate axes.

3.2.1.1 DIRECTION COSINE MATRIX FROM TRANSFORMED VECTOR COMPONENTS

Occasionally the components of a vector are known in two coordinate frames and it is desired to calculate the direction cosine matrix between the two frames from the vector components. Specifically, the problem we pose in this section is: Given $\underline{V}^{A_1}$ and $\underline{V}^{A_2}$, Find $C_{A_2}^{A_1}$ that satisfies:

$$\underline{V}^{A_1} = C_{A_2}^{A_1} \underline{V}^{A_2} \tag{3.2.1.1-1}$$

There is no unique solution for $C_{A_2}^{A_1}$ that can be obtained from Equation (3.2.1.1-1) without imposing an additional constraint. A common form of the constraint is the equivalent to Equation (3.2.1.1-1) operating on another vector:

$$\underline{U}^{A_1} = C_{A_2}^{A_1} \underline{U}^{A_2} \tag{3.2.1.1-2}$$

where

$\underline{U}$ = Another vector.

We also define a third vector $\underline{W}$ as the cross-product between $\underline{U}$ and $\underline{V}$. In the A_2 Frame:

$$\underline{W}^{A_2} \equiv \underline{U}^{A_2} \times \underline{V}^{A_2} \tag{3.2.1.1-3}$$

The A_2 Frame components of $\underline{W}$ can be expressed as a function of the A_1 Frame components as:

$$\underline{W}^{A_1} = C_{A_2}^{A_1} \underline{W}^{A_2} \tag{3.2.1.1-4}$$

We can also calculate $\underline{W}^{A_1}$ using generalized Equation (3.1.1-42):

$$\underline{W}^{A_1} = \underline{U}^{A_1} \times \underline{V}^{A_1} \tag{3.2.1.1-5}$$

Given $\underline{U}^{A_1}$, $\underline{V}^{A_1}$, $\underline{U}^{A_2}$ and $\underline{V}^{A_2}$, Equations (3.2.1.1-1), (3.2.1.1-2) and (3.2.1.1-4) with (3.2.1.1-3) and (3.2.1.1-5) constitute a deterministic set of simultaneous equations that can now be solved for $C_{A_2}^{A_1}$. The solution is obtained by first defining the following matrix forms:

$$D^{A_1} \equiv \left[\underline{U}^{A_1} \;\; \underline{V}^{A_1} \;\; \underline{W}^{A_1} \right] \qquad\qquad D^{A_2} \equiv \left[\underline{U}^{A_2} \;\; \underline{V}^{A_2} \;\; \underline{W}^{A_2} \right] \tag{3.2.1.1-6}$$

for which Equations (3.2.1.1-1), (3.2.1.1-2) and (3.2.1.1-4) become:

$$D^{A_1} = C_{A_2}^{A_1} D^{A_2} \tag{3.2.1.1-7}$$

The solution to (3.2.1.1-7) for $C_{A_2}^{A_1}$ then is:

$$C_{A_2}^{A_1} = D^{A_1} \left(D^{A_2}\right)^{-1} \tag{3.2.1.1-8}$$

Equation (3.2.1.1-8) is valid provided that $\left(D^{A_2}\right)^{-1}$ is non-singular. The non-singularity condition is that the determinant of D^{A_2} be non-zero. By substituting (3.2.1.1-3) into the (3.2.1.1-6) D^{A_2} expression and expanding $\underline{U}^{A_2}$, $\underline{V}^{A_2}$, $\underline{W}^{A_2}$ in A_2 Frame component form, it is easily shown that the determinant of D^{A_2} is:

$$\text{Determinant}\left(D^{A_2}\right) = \underline{W}^{A_2} \cdot \underline{W}^{A_2} \tag{3.2.1.1-9}$$

Thus, the D^{A2} determinant is non-zero (and $\left(D^{A2}\right)^{-1}$ non-singular) if $\underline{W}^{A2}$ is non-zero. From Equation (3.2.1.1-3) we see that this condition is satisfied if $\underline{U}^{A2}$ and $\underline{V}^{A2}$ are non-parallel. An optimum configuration would have $\underline{U}^{A2}$ perpendicular to $\underline{V}^{A2}$.

Let us now consider another solution approach for $C_{A_2}^{A_1}$ in Equation (3.2.1.1-1) based on a different constraint relationship. To do this, we employ the "rotation vector" concept (to be introduced in Section 3.2.2) in which the attitude of coordinate Frame A_2 is defined relative to Frame A_1 as the attitude that Frame A_1 would assume following a single rotation of Frame A_1 around a specified "rotation vector" through an angle equal to the rotation vector magnitude. Generalized Equation (3.2.2.1-4) defines the equivalency between the $C_{A_2}^{A_1}$ direction cosine matrix and its associated rotation vector. Let's see what the solution is for $C_{A_2}^{A_1}$ in Equation (3.2.1.1-1), if we define a constraint relationship for (3.2.1.1-1) as the requirement that the magnitude of the rotation vector associated with $C_{A_2}^{A_1}$ be minimum. To embody this constraint, we begin by substituting generalized Equation (3.2.2.1-4) with (3.2.2.1-6) for $C_{A_2}^{A_1}$ into (3.2.1.1-1) using (3.1.1-15):

$$\underline{V}^{A1} = \underline{V}^{A2} + \sin\phi\left(\underline{u}_\phi^{A2} \times \underline{V}^{A2}\right) + (1 - \cos\phi)\,\underline{u}_\phi^{A2} \times \left(\underline{u}_\phi^{A2} \times \underline{V}^{A2}\right) \qquad (3.2.1.1\text{-}10)$$

where

$\underline{u}_\phi^{A2}$ = Unit vector along the rotation vector associated with $C_{A_2}^{A_1}$.

ϕ = Magnitude of the $C_{A_2}^{A_1}$ rotation vector.

The triple cross-product term in (3.2.1.1-10) can be expanded using the Equation (3.1.1-16) triple vector product identity:

$$\underline{u}_\phi^{A2} \times \left(\underline{u}_\phi^{A2} \times \underline{V}^{A2}\right) = \underline{u}_\phi^{A2}\left(\underline{u}_\phi^{A2} \cdot \underline{V}^{A2}\right) - \underline{V}^{A2} \qquad (3.2.1.1\text{-}11)$$

Substituting (3.2.1.1-11) in (3.2.1.1-10) then yields:

$$\underline{V}^{A1} = \sin\phi\left(\underline{u}_\phi^{A2} \times \underline{V}^{A2}\right) + \cos\phi\,\underline{V}^{A2} + (1 - \cos\phi)\,V_\phi\,\underline{u}_\phi^{A2} \qquad (3.2.1.1\text{-}12)$$

with

$$V_\phi \equiv \underline{u}_\phi^{A2} \cdot \underline{V}^{A2} \qquad (3.2.1.1\text{-}13)$$

where

V_ϕ = Component of $\underline{V}^{A2}$ along $\underline{u}_\phi^{A2}$.

If (3.2.1.1-12) is now multiplied by the transpose of $\underline{V}^{A2}$, we find after application of (3.1.1-10), substitution of (3.2.1.1-13) and application of generalized Equation (3.1.1-2) to the dot product of $\underline{V}^{A2}$ with itself, that:

$$\begin{aligned}\left(\underline{V}^{A2}\right)^T \underline{V}^{A1} &= \sin\phi \left(\underline{V}^{A2}\right)^T \left(\underline{u}_\phi^{A2} \times \underline{V}^{A2}\right) + \cos\phi \left(\underline{V}^{A2}\right)^T \underline{V}^{A2} \\ &\qquad + (1 - \cos\phi)\, V_\phi \left(\underline{V}^{A2}\right)^T \underline{u}_\phi^{A2} \\ &= \sin\phi\, \underline{V}^{A2} \cdot \left(\underline{u}_\phi^{A2} \times \underline{V}^{A2}\right) + \cos\phi\, \underline{V}^{A2} \cdot \underline{V}^{A2} \\ &\qquad + (1 - \cos\phi)\, V_\phi\, \underline{V}^{A2} \cdot \underline{u}_\phi^{A2} \\ &= V^2 \cos\phi + (1 - \cos\phi)\, V_\phi^2 = \left(V^2 - V_\phi^{\,2}\right)\cos\phi + V_\phi^2\end{aligned} \tag{3.2.1.1-14}$$

or upon rearrangement:

$$\cos\phi = \frac{\left(\underline{V}^{A2}\right)^T \underline{V}^{A1} - V_\phi^{\,2}}{V^2 - V_\phi^{\,2}} \tag{3.2.1.1-15}$$

where

V = Magnitude of $\underline{V}$ (i.e., the magnitude of $\underline{V}^{A2}$ or $\underline{V}^{A1}$).

Equation (3.2.1.1-15) can now be used to apply our selected constraint of minimizing ϕ. We define the minimization problem as finding V_ϕ that minimizes ϕ which is equivalent to maximizing $\cos\phi$. The result is readily achieved by equating the derivative of (3.2.1.1-15) to zero:

$$\begin{aligned}\frac{d\cos\phi}{dV_\phi} &= \frac{-2\left(V^2 - V_\phi^{\,2}\right)V_\phi + 2\left[\left(\underline{V}^{A2}\right)^T \underline{V}^{A1} - V_\phi^{\,2}\right]V_\phi}{\left(V^2 - V_\phi^{\,2}\right)^2} \\ &= \frac{2\left[\left(\underline{V}^{A2}\right)^T \underline{V}^{A1} - V^2\right]V_\phi}{\left(V^2 - V_\phi^{\,2}\right)^2} = 0\end{aligned} \tag{3.2.1.1-16}$$

from which:

$$V_\phi = 0 \quad \text{For Minimum } \phi \tag{3.2.1.1-17}$$

Substituting (3.2.1.1-17) into (3.2.1.1-15) and (3.2.1.1-12) we find:

$$\cos\phi = \frac{1}{V^2}\left(\underline{V}^{A2}\right)^T \underline{V}^{A1} \quad \text{For Minimum } \phi \tag{3.2.1.1-18}$$

$$\underline{V}^{A1} = \sin\phi\left(\underline{u}_{\phi}^{A2} \times \underline{V}^{A2}\right) + \cos\phi\, \underline{V}^{A2} \quad \text{For Minimum } \phi \tag{3.2.1.1-19}$$

Multiplying (3.2.1.1-19) by the $\underline{V}^{A2}$ cross-product operator, applying (3.1.1-13) on the right with the (3.1.1-16) triple vector product identity, and using (3.2.1.1-17) with (3.2.1.1-13) gives:

$$\begin{aligned}\left(\underline{V}^{A2}\times\right)\underline{V}^{A1} &= \sin\phi\, \underline{V}^{A2} \times \left(\underline{u}_{\phi}^{A2} \times \underline{V}^{A2}\right) + \cos\phi\, \underline{V}^{A2} \times \underline{V}^{A2} \\ &= \sin\phi\, \underline{V}^{A2} \times \left(\underline{u}_{\phi}^{A2} \times \underline{V}^{A2}\right) \\ &= \sin\phi\left[\underline{u}_{\phi}^{A2}\left(\underline{V}^{A2} \cdot \underline{V}^{A2}\right) - \underline{V}^{A2}\left(\underline{u}_{\phi}^{A2} \cdot \underline{V}^{A2}\right)\right] \\ &= \sin\phi\, \underline{u}_{\phi}^{A2}\, V^2 \qquad \text{For Minimum } \phi\end{aligned} \tag{3.2.1.1-20}$$

from which we obtain:

$$\sin\phi\, \underline{u}_{\phi}^{A2} = \frac{1}{V^2}\left(\underline{V}^{A2}\times\right)\underline{V}^{A1} \quad \text{For Minimum } \phi \tag{3.2.1.1-21}$$

Equations (3.2.1.1-18) and (3.2.1.1-21) can now be used to construct the $C_{A_2}^{A_1}$ matrix for minimum ϕ. First we introduce the following definitions for the vector terms in these equations:

$$D \equiv \frac{1}{V^2}\left(\underline{V}^{A2}\right)^T \underline{V}^{A1} \qquad \underline{E} \equiv \frac{1}{V^2}\left(\underline{V}^{A2}\times\right)\underline{V}^{A1} \tag{3.2.1.1-22}$$

Then we note from (3.2.1.1-21) and (3.2.1.1-22) that:

$$\left(\underline{E}\times\right)^2 = \sin^2\phi\left(\underline{u}_{\phi}^{A2}\times\right)^2 \tag{3.2.1.1-23}$$

The $\sin^2\phi$ term in (3.2.1.1-23) is expanded using (3.2.1.1-18) and (3.2.1.1-22) as:

$$\sin^2\phi = 1 - \cos^2\phi = \left(1 - \cos\phi\right)\left(1 + \cos\phi\right) = \left(1 - \cos\phi\right)\left(1 + D\right) \tag{3.2.1.1-24}$$

from which (3.2.1.1-23) becomes:

$$\left(\underline{u}_{\phi}^{A2}\times\right)^2 = \frac{1}{\left(1 - \cos\phi\right)\left(1 + D\right)}\left(\underline{E}\times\right)^2 \tag{3.2.1.1-25}$$

Finally, the solution for $C_{A_2}^{A_1}$ is obtained from generalized Equation (3.2.2.1-4) with (3.2.2.1-6) by substituting (3.2.1.1-25) and (3.2.1.1-21) for corresponding terms using (3.2.1.1-22) for D and $\underline{E}$:

$$C_{A_2}^{A_1} = I + \left(\underline{E}\times\right) + \frac{1}{1 + D}\left(\underline{E}\times\right)^2 \quad \text{For Minimum } \phi \tag{3.2.1.1-26}$$

3.2.2 ROTATION VECTOR

Another way of describing the attitude of an arbitrary coordinate Frame B relative to another arbitrary coordinate Frame A is through the "rotation vector" concept. The "rotation vector" defines an axis of rotation and magnitude for a rotation about the rotation vector (using the standard right hand convention for rotation about a vector). Imagine Frame A being rotated from its starting attitude to a new attitude by rotation about the "rotation vector" through an angle equal to the rotation vector magnitude. Now call Frame B the new attitude of Frame A. By this definition of Frame B, an arbitrarily defined rotation vector uniquely defines the attitude of Frame B relative to the original Frame A attitude. Conversely, for a given Frame B attitude relative to Frame A, a rotation vector can be defined that is consistent with this attitude. Thus a rotation vector can be used to define the attitude of Frame B relative to Frame A.

From an analytical standpoint, consider the Frame B coordinate frame I-axis unit vector $\underline{u}_{IB}$ (I for coordinate axes X, Y, or Z - See Section 3.1 for definition) and how it looks in Frame A after it has been rotated from the Frame A attitude around the rotation vector into the Frame B attitude. For this analysis, let us first define:

$$\underline{\phi} = \phi\, \underline{u}_{\phi} \qquad \phi = \sqrt{\underline{\phi} \cdot \underline{\phi}} \qquad \underline{u}_{\phi} = \underline{\phi} / \phi \tag{3.2.2-1}$$

where

ϕ = Magnitude of the rotation vector.

$\underline{u}_{\phi}$ = Unit vector in the rotation vector direction.

$\underline{\phi}$ = Rotation vector.

We consider Frame B to be "initially" aligned with Frame A, and then to be rotated around the rotation vector from its initial attitude into its "final" Frame B attitude. From the previous definitions, application of the rotation vector to coordinate Frame B then is to rotate each of the Frame B coordinate i axis unit vectors ($\underline{u}_{iB}$) from Frame A about $\underline{u}_{\phi}$ through angle ϕ into the "final" Frame B axis orientation. Figure 3.2.2-1 describes the geometry from the viewpoint of Frame A:

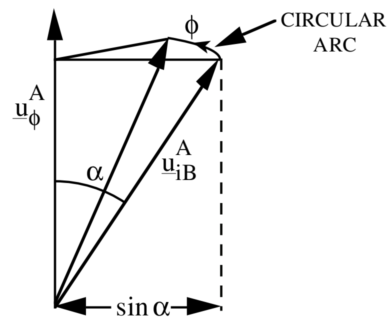


Figure 3.2.2-1 The Geometry From The Viewpoint Of Frame A

From the viewpoint of Frame A as depicted in Figure 3.2.2-1, $\underline{u}_{iB}^{A}$ traces a circular cone around $\underline{u}_{\phi}^{A}$ as it is rotated through rotation angle ϕ from its original Frame A attitude into its final B Frame attitude. Let us consider this process as a sequence of infinitesimal rotations $d\phi$ around $\underline{u}_{\phi}^{A}$. Each $d\phi$ produces $d\underline{u}_{iB}^{A}$, an infinitesimal change in $\underline{u}_{iB}^{A}$ that is tangent to the ϕ circular arc in Figure 3.2.2-1 in the direction shown, whose magnitude equals $d\phi$ times the component of $\underline{u}_{iB}^{A}$ perpendicular to $\underline{u}_{\phi}^{A}$ (i.e., the circular arc radius). As shown in Figure 3.2.2-1, the circular arc radius equals the sine of the angle between $\underline{u}_{\phi}^{A}$ and $\underline{u}_{iB}^{A}$. Hence:

$$\left| d\underline{u}_{iB}^{A} \right| = d\phi \left| \sin\alpha \right| \tag{3.2.2-2}$$

From the (3.1.1-3) definition of the cross-product between two vectors (and the notes following Equation (3.1.1-3)), the previous magnitude and direction properties of $d\underline{u}_{iB}^{A}$ show that:

$$d\underline{u}_{iB}^{A} = d\phi \; \underline{u}_{\phi}^{A} \times \underline{u}_{iB}^{A} \tag{3.2.2-3}$$

or equivalently, with (3.1.1-13):

$$\frac{d}{d\phi} \underline{u}_{iB}^{A} = \underline{u}_{\phi}^{A} \times \underline{u}_{iB}^{A} = \left(\underline{u}_{\phi}^{A} \times \right) \underline{u}_{iB}^{A} \tag{3.2.2-4}$$

Recognizing $\underline{u}_{\phi}^{A}$ as constant, successive differentiation of (3.2.2-4) with respect to ϕ and substitution of (3.2.2-4) gives:

$$\frac{d^2}{d\phi^2}\,\underline{u}_{iB}^{A} = \left(\underline{u}_{\phi}^{A}\times\right)\frac{d}{d\phi}\,\underline{u}_{iB}^{A} = \left(\underline{u}_{\phi}^{A}\times\right)^2\underline{u}_{iB}^{A} \tag{3.2.2-5}$$

$$\frac{d^3}{d\phi^3}\,\underline{u}_{iB}^{A} = \left(\underline{u}_{\phi}^{A}\times\right)^2\frac{d}{d\phi}\,\underline{u}_{iB}^{A} = \left(\underline{u}_{\phi}^{A}\times\right)^3\underline{u}_{iB}^{A} \tag{3.2.2-6}$$

Using (3.1.1-15), the general Equation (3.1.1-16) vector triple cross-product identity can be written as:

$$\left(\underline{V}_1\times\right)\left(\underline{V}_2\times\right)\underline{V}_3 = \underline{V}_2\left(\underline{V}_1\cdot\underline{V}_3\right) - \underline{V}_3\left(\underline{V}_1\cdot\underline{V}_2\right) \tag{3.2.2-7}$$

For $\underline{V}_1 = \underline{V}_2 = \underline{V}$, the cross-product of $\underline{V}$ with (3.2.2-7) and application of (3.1.1-15) shows that for arbitrary $\underline{V}$ and $\underline{V}_3$:

$$\left(\underline{V}\times\right)\left[\left(\underline{V}\times\right)^2\underline{V}_3\right] = -V^2\left(\underline{V}\times\right)\underline{V}_3$$

or

$$\left(\underline{V}\times\right)^3\underline{V}_3 = -V^2\left(\underline{V}\times\right)\underline{V}_3$$

or

$$\left(\underline{V}\times\right)^3 = -V^2\left(\underline{V}\times\right) \tag{3.2.2-8}$$

where

V = Magnitude of $\underline{V}$.

With (3.2.2-8) and $\underline{u}_{\phi}^{A}$ recognized as a unit vector, Equation (3.2.2-6) becomes with (3.2.2-4):

$$\frac{d^3}{d\phi^3}\,\underline{u}_{iB}^{A} = -\left(\underline{u}_{\phi}^{A}\times\right)\underline{u}_{iB}^{A} = -\frac{d}{d\phi}\,\underline{u}_{iB}^{A} \tag{3.2.2-9}$$

or

$$\frac{d^3}{d\phi^3}\,\underline{u}_{iB}^{A} + \frac{d}{d\phi}\,\underline{u}_{iB}^{A} = 0 \tag{3.2.2-10}$$

Equation (3.2.2-10) is a linear homogeneous constant coefficient differential equation with respect to ϕ for $\underline{u}_{iB}^{A}$ whose general solution has the form $L\,e^{\lambda\phi}$ where:

λ = Characteristic root of (3.2.2-10).

L = Constant dependent on initial conditions.

Substitution of $L\, e^{\lambda \phi}$ for $\underline{u}_{iB}^{A}$ in (3.2.2-10) provides the characteristic equation for λ:

$$\lambda^3 + \lambda = 0 \qquad (3.2.2\text{-}11)$$

whose solutions are:

$$\lambda = 0, \pm j \qquad (3.2.2\text{-}12)$$

where

j = Imaginary parameter equal to $\sqrt{-1}$.

The general $L\, e^{\lambda \phi}$ solution to (3.2.2-10) using the (3.2.2-12) roots has the form:

$$\underline{u}_{iB}^{A} = \underline{L}_0 + \underline{L}_1\, e^{j\phi} + \underline{L}_2\, e^{-j\phi} \qquad (3.2.2\text{-}13)$$

where

$\underline{L}_0, \underline{L}_1, \underline{L}_2$ = Constant vectors set to fit initial conditions.

Euler's theorem states that:

$$e^{j\, x} = \cos x + j \sin x \qquad (3.2.2\text{-}14)$$

where

x = An arbitrary real number.

Equation (3.2.2-14) is easily proven by substituting the Taylor series expansions for $\sin x$ and $\cos x$ in $\cos x + j \sin x$ and seeing that the result corresponds to the Taylor series expansion for $e^{j\, x}$.

Applying (3.2.2-14) in (3.2.2-13) yields the alternative form for the general $\underline{u}_{iB}^{A}$ solution:

$$\underline{u}_{iB}^{A} = \underline{B}_0 + \underline{B}_s \sin \phi + \underline{B}_c \cos \phi \qquad (3.2.2\text{-}15)$$

where

$\underline{B}_0, \underline{B}_s, \underline{B}_c$ = Constant vectors set to fit initial conditions.

The $\underline{B}_0, \underline{B}_s, \underline{B}_c$ vectors are determined from (3.2.2-15) and its derivatives as follows. The derivatives are:

$$\frac{d}{d\phi}\,\underline{u}_{iB}^{A} \;=\; \underline{B}_s \cos\phi - \underline{B}_c \sin\phi \qquad\qquad \frac{d^2}{d\phi^2}\,\underline{u}_{iB}^{A} \;=\; -\,\underline{B}_s \sin\phi - \underline{B}_c \cos\phi \qquad (3.2.2\text{-}16)$$

At the start of the ϕ rotation (i.e., at $\phi = 0$), $\underline{u}_{IB}$ is aligned with A Frame axis i so that:

$$\underline{u}_{iB}^{A} \;=\; \underline{u}_{iA}^{A} \qquad \text{At } \phi = 0 \qquad (3.2.2\text{-}17)$$

Equating (3.2.2-16) to (3.2.2-4) and (3.2.2-5) at $\phi = 0$, setting $\phi = 0$ in Equation (3.2.2-15) and applying Equation (3.2.2-17) generates three simultaneous equations for $\underline{B}_0$, $\underline{B}_s$, $\underline{B}_c$:

$$\underline{u}_{iA}^{A} \;=\; \underline{B}_0 + \underline{B}_c \qquad \left(\underline{u}_{\phi}^{A}\times\right)\underline{u}_{iA}^{A} \;=\; \underline{B}_s \qquad \left(\underline{u}_{\phi}^{A}\times\right)^2 \underline{u}_{iA}^{A} \;=\; -\,\underline{B}_c \qquad (3.2.2\text{-}18)$$

From (3.2.2-18), the values for $\underline{B}_0$, $\underline{B}_s$, $\underline{B}_c$ are:

$$\underline{B}_0 \;=\; \left[I + \left(\underline{u}_{\phi}^{A}\times\right)^2\right]\underline{u}_{iA}^{A} \qquad \underline{B}_s \;=\; \left(\underline{u}_{\phi}^{A}\times\right)\underline{u}_{iA}^{A} \qquad \underline{B}_c \;=\; -\left(\underline{u}_{\phi}^{A}\times\right)^2 \underline{u}_{iA}^{A} \qquad (3.2.2\text{-}19)$$

Substituting (3.2.2-19) into (3.2.2-15) then provides the desired result for $\underline{u}_{iB}^{A}$:

$$\underline{u}_{iB}^{A} \;=\; \left[I + \sin\phi\left(\underline{u}_{\phi}^{A}\times\right) + (1 - \cos\phi)\left(\underline{u}_{\phi}^{A}\times\right)\left(\underline{u}_{\phi}^{A}\times\right)\right]\underline{u}_{iA}^{A} \qquad (3.2.2\text{-}20)$$

Equation (3.2.2-20) shows how the i^{th} Frame B coordinate axis unit vector is related to the i^{th} Frame A coordinate axis unit vector as a function of the rotation vector $\underline{\phi}$.

3.2.2.1 DIRECTION COSINE MATRIX IN TERMS OF ROTATION VECTOR

Equation (3.2.2-20) is the basis for the equivalency equation relating the direction cosine matrix to the rotation vector. Recall from Equation (3.2.1-6) that:

$$C_B^A \;=\; \left[\; \underline{u}_{1B}^{A} \;\; \underline{u}_{2B}^{A} \;\; \underline{u}_{3B}^{A} \;\right] \qquad (3.2.2.1\text{-}1)$$

Substituting (3.2.2-20) into Equation (3.2.2.1-1) yields:

$$C_B^A \;=\; \left[I + \sin\phi\left(\underline{u}_{\phi}^{A}\times\right) + (1 - \cos\phi)\left(\underline{u}_{\phi}^{A}\times\right)\left(\underline{u}_{\phi}^{A}\times\right)\right]\left[\underline{u}_{1A}^{A}\;\underline{u}_{2A}^{A}\;\underline{u}_{3A}^{A}\right] \qquad (3.2.2.1\text{-}2)$$

But note that:

$$\underline{u}_{1A}^{A} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \underline{u}_{2A}^{A} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \underline{u}_{3A}^{A} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{3.2.2.1-3}$$

Equations (3.2.2.1-3) substituted in (3.2.2.1-2) provides an expression for the direction cosine matrix as a function of the rotation vector:

$$C_B^A = I + \sin\phi \left(\underline{u}_\phi^A \times\right) + (1 - \cos\ \phi) \left(\underline{u}_\phi^A \times\right)\left(\underline{u}_\phi^A \times\right) \tag{3.2.2.1-4}$$

Equation (3.2.2.1-4) can be generalized further by first taking its transpose and observing that the transpose of a cross-product operator equals its negative. Applying Equation (3.2.1-3) then shows that:

$$C_A^B = \left(C_B^A\right)^T = I - \sin\phi \left(\underline{u}_\phi^A \times\right) + (1 - \cos\ \phi) \left(\underline{u}_\phi^A \times\right)\left(\underline{u}_\phi^A \times\right) \tag{3.2.2.1-5}$$

Now let's transform the rotation vector unit vector from Frame A ($\underline{u}_\phi^A$) to Frame B (i.e., to obtain $\underline{u}_\phi^B$) using Equation (3.2.2.1-5). The result is:

$$\underline{u}_\phi^B = \underline{u}_\phi^A \tag{3.2.2.1-6}$$

Then with Equations (3.2.2-1):

$$\underline{\phi}^B = \underline{\phi}^A \tag{3.2.2.1-7}$$

Equations (3.2.2.1-6) and (3.2.2.1-7) state the fundamental property of the rotation vector's unit vector and the rotation vector itself; each has identical components in Frames A and B. This should be obvious geometrically from the definition of the rotation vector which obtains Frame B by rotation of Frame A about the rotation vector. Without loss of generality, we can therefore, drop the superscript notation from the rotation vector. With this simplification and application of Equations (3.2.2-1) to (3.2.2.1-4) for $\underline{u}_\phi$ as a function of $\underline{\phi}$ and its magnitude, we obtain the final expression for the direction cosine matrix in terms of the rotation vector:

$$C_B^A = I + \frac{\sin\ \phi}{\phi}\left(\underline{\phi}\times\right) + \frac{(1 - \cos\ \phi)}{\phi^2}\left(\underline{\phi}\times\right)\left(\underline{\phi}\times\right) \tag{3.2.2.1-8}$$

The trigonometric coefficients in Equation (3.2.2.1-8) can be easily evaluated without singularities from the following equivalent Taylor series expansion formulas:

$$\frac{\sin\phi}{\phi} = 1 - \frac{\phi^2}{3\,!} + \frac{\phi^4}{5\,!} - \cdots \qquad \frac{(1 - \cos\phi)}{\phi^2} = \frac{1}{2\,!} - \frac{\phi^2}{4\,!} + \frac{\phi^4}{6\,!} - \cdots \tag{3.2.2.1-9}$$

3.2.2.2 ROTATION VECTOR IN TERMS OF DIRECTION COSINES

The converse of Equation (3.2.2.1-8) can be derived by returning to Equations (3.2.2.1-4) and (3.2.2.1-5), dropping the superscript on $\underline{u}_\phi^A$ for simplicity (without loss of generality as discussed in Section 3.2.2.1):

$$C_B^A = I + \sin\phi\,(\underline{u}_\phi\times) + (1 - \cos\phi)\,(\underline{u}_\phi\times)(\underline{u}_\phi\times) \tag{3.2.2.2-1}$$

$$\left(C_B^A\right)^T = I - \sin\phi\,(\underline{u}_\phi\times) + (1 - \cos\phi)\,(\underline{u}_\phi\times)(\underline{u}_\phi\times) \tag{3.2.2.2-2}$$

Let us define two intermediate matrices G and H as follows:

$$G \equiv \frac{1}{2}\left[C_B^A - \left(C_B^A\right)^T\right] \qquad H \equiv \frac{1}{2}\left[C_B^A + \left(C_B^A\right)^T\right] \tag{3.2.2.2-3}$$

Substituting Equations (3.2.2.2-1) and (3.2.2.2-2) into (3.2.2.2-3) obtains:

$$G = \sin\phi\,(\underline{u}_\phi\times) \tag{3.2.2.2-4}$$

$$H = I + (1 - \cos\phi)\,(\underline{u}_\phi\times)(\underline{u}_\phi\times) \tag{3.2.2.2-5}$$

Let us now define the components of C_B^A (as in (3.2.1-1)) and $\underline{u}_\phi$ as follows:

$$\underline{u}_\phi = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \qquad C_B^A = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \tag{3.2.2.2-6}$$

From its definition as a unit vector, we can write for the components of $\underline{u}_\phi$:

$$u_1{}^2 + u_2{}^2 + u_3{}^2 = 1 \tag{3.2.2.2-7}$$

Substituting (3.2.2.2-6) into (3.2.2.2-3) through (3.2.2.2-5) and applying (3.2.2.2-7):

$$G = \frac{1}{2}\begin{bmatrix} 0 & (C_{12} - C_{21}) & (C_{13} - C_{31}) \\ (C_{21} - C_{12}) & 0 & (C_{23} - C_{32}) \\ (C_{31} - C_{13}) & (C_{32} - C_{23}) & 0 \end{bmatrix} = \sin\phi \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \qquad (3.2.2.2\text{-}8)$$

$$H = \frac{1}{2}\begin{bmatrix} 2\,C_{11} & (C_{12} + C_{21}) & (C_{13} + C_{31}) \\ (C_{21} + C_{12}) & 2\,C_{22} & (C_{23} + C_{32}) \\ (C_{31} + C_{13}) & (C_{32} + C_{23}) & 2\,C_{33} \end{bmatrix}$$

$$= \begin{bmatrix} 1 + (1 - \cos\phi)(u_1^2 - 1) & (1 - \cos\phi)\, u_1\, u_2 & (1 - \cos\phi)\, u_1\, u_3 \\ (1 - \cos\phi)\, u_2\, u_1 & 1 + (1 - \cos\phi)(u_2^2 - 1) & (1 - \cos\phi)\, u_2\, u_3 \\ (1 - \cos\phi)\, u_3\, u_1 & (1 - \cos\phi)\, u_3\, u_2 & 1 + (1 - \cos\phi)(u_3^2 - 1) \end{bmatrix} \qquad (3.2.2.2\text{-}9)$$

Summing the square of the $\underline{u}_\phi$ component elements in Equation (3.2.2.2-8), applying (3.2.2.2-7) and taking the square root yields:

$$\sin\phi = \frac{1}{2}\sqrt{(C_{32} - C_{23})^2 + (C_{13} - C_{31})^2 + (C_{21} - C_{12})^2} \qquad (3.2.2.2\text{-}10)$$

Selecting the positive sign for the square root in Equation (3.2.2.2-10) identifies ϕ as a positive rotation between 0 and π (without loss in generality since ϕ is defined as a magnitude quantity. The case when ϕ is greater than π (but less than $2\,\pi$) is handled by treating ϕ as $2\,\pi - \phi$ with $\underline{u}_\phi$ defined to be in the opposite direction (i.e., the attitude resulting from a rotation of ϕ about $\underline{u}_\phi$ is identical to the attitude resulting from a rotation of $2\,\pi - \phi$ about $-\underline{u}_\phi$).

Summing the diagonal elements of Equation (3.2.2.2-9) and applying (3.2.2.2-7) yields:

$$\cos\phi = \frac{1}{2}(C_{11} + C_{22} + C_{33} - 1) \qquad (3.2.2.2\text{-}11)$$

With (3.2.2.2-10) and (3.2.2.2-11) we then obtain for ϕ:

$$\phi = \tan^{-1}\frac{\sin\phi}{\cos\phi} = \tan^{-1}\left(\frac{\sqrt{(C_{32} - C_{23})^2 + (C_{13} - C_{31})^2 + (C_{21} - C_{12})^2}}{(C_{11} + C_{22} + C_{33} - 1)}\right) \qquad (3.2.2.2\text{-}12)$$

The components of $\underline{\phi}$ are determined from one of two forms of the $\underline{\phi}$ expression in Equations (3.2.2-1), depending on whether ϕ is greater or less than $\pi/2$ (i.e., $\cos\phi$ is less than or greater than 0):

$$\underline{\phi} = \frac{\phi}{\sin\phi} \sin\phi \, \underline{u}_\phi \quad \text{For } \cos\phi \geq 0 \tag{3.2.2.2-13}$$

$$\underline{\phi} = \phi \, \underline{u}_\phi \quad \text{For } \cos\phi < 0 \tag{3.2.2.2-14}$$

For the Equation (3.2.2.2-13) situation we first write the Taylor series expansion for the ratio of $\sin\phi$ over ϕ as:

$$f \equiv \frac{\sin\phi}{\phi} = 1 - \frac{\phi^2}{3\,!} + \frac{\phi^4}{5\,!} - \frac{\phi^6}{7\,!} + \cdots \tag{3.2.2.2-15}$$

The reciprocal of Equation (3.2.2.2-15) for Equation (3.2.2.2-13) is defined as:

$$F \equiv \frac{\phi}{\sin\phi} = \frac{1}{f} \tag{3.2.2.2-16}$$

We then use F from Equations (3.2.2.2-15) and (3.2.2.2-16) in Equation (3.2.2.2-13) with the components of $\sin\phi \, \underline{u}_\phi$ from Equation (3.2.2.2-8) to obtain:

For $\cos\phi \geq 0$:

$$\phi_X = \frac{1}{2} F \left(C_{32} - C_{23}\right) \qquad \phi_Y = \frac{1}{2} F \left(C_{13} - C_{31}\right) \qquad \phi_Z = \frac{1}{2} F \left(C_{21} - C_{12}\right) \tag{3.2.2.2-17}$$

where

ϕ_X, ϕ_Y, ϕ_Z = X, Y, Z components of $\underline{\phi}$.

Note from the (3.2.2.2-15) definition for f, that f goes to zero at $\phi = \pi$ which would make F infinite, thereby producing a singularity in the Equation (3.2.2.2-17) set. This is the reason that Equations (3.2.2.2-17) are restricted to use for $\cos\phi \geq 0$ (i.e., $0 \leq \phi \leq \pi / 2$).

For the situation when Equation (3.2.2.2-14) applies, we first compute the magnitude of the $\underline{u}_\phi$ components from the diagonal elements in Equation (3.2.2.2-9) and find the maximum of the three:

For $\cos\phi < 0$:

$$|u_1| = \sqrt{\frac{C_{11} - 1}{1 - \cos\phi} + 1} \qquad |u_2| = \sqrt{\frac{C_{22} - 1}{1 - \cos\phi} + 1} \qquad |u_3| = \sqrt{\frac{C_{33} - 1}{1 - \cos\phi} + 1}$$

$$u_{MAX} = \max\left(|u_1|, |u_2|, |u_3|\right) \tag{3.2.2.2-18}$$

where

u_{MAX} = Largest of $|u_1|$, $|u_2|$ and $|u_3|$.

For the maximum magnitude component of $\underline{u}_\phi$ from Equations (3.2.2.2-18), we calculate the sign from the appropriate off-diagonal element of the G matrix in Equation (3.2.2.2-8), and then compute the remaining elements of $\underline{u}_\phi$ from the appropriate off-diagonal elements in Equation (3.2.2.2-9) using the maximum component with sign (calculated as previously described). The components of $\underline{\phi}$ are then determined from Equation (3.2.2.2-14) using ϕ from Equation (3.2.2.2-12), $\cos\phi$ from Equation (3.2.2.2-11), and the previously described computed components of $\underline{u}_\phi$, whence:

For $\cos\phi < 0$:

If $|u_1| = u_{MAX}$, then:

$$u_1 = |u_1| \,\mathrm{Sign}\left(C_{32} - C_{23}\right) \qquad u_2 = \frac{1}{2\,u_1}\,\frac{C_{12} + C_{21}}{1 - \cos\phi} \qquad u_3 = \frac{1}{2\,u_1}\,\frac{C_{13} + C_{31}}{1 - \cos\phi}$$

If $|u_2| = u_{MAX}$, then:

$$u_2 = |u_2| \,\mathrm{Sign}\left(C_{13} - C_{31}\right) \qquad u_3 = \frac{1}{2\,u_2}\,\frac{C_{23} + C_{32}}{1 - \cos\phi} \qquad u_1 = \frac{1}{2\,u_2}\,\frac{C_{12} + C_{21}}{1 - \cos\phi} \tag{3.2.2.2-19}$$

If $|u_3| = u_{MAX}$, then:

$$u_3 = |u_3| \,\mathrm{Sign}\left(C_{21} - C_{12}\right) \qquad u_1 = \frac{1}{2\,u_3}\,\frac{C_{13} + C_{31}}{1 - \cos\phi} \qquad u_2 = \frac{1}{2\,u_3}\,\frac{C_{23} + C_{32}}{1 - \cos\phi}$$

$$\phi_X = \phi\, u_1 \qquad \phi_Y = \phi\, u_2 \qquad \phi_Z = \phi\, u_3$$

where

Sign ( ) = +1 if ( ) is ≥ 0, and -1 if ( ) is < 0.

Note in Equations (3.2.2.2-19) that calculation of the second and third $\underline{u}_\phi$ components entails a division by the first. Selecting the first as the maximum of the three assures that the singularity of division by zero will not occur in computing the remaining two. Choosing the maximum also assures that amplification of C_{ij} errors will be minimized in the (3.2.2.2-19) $\underline{u}_\phi$ component calculations. (A classical source of C_{ij} error in strapdown inertial navigation systems is produced in the computation of the C_{ij}'s using angular rate sensor inputs containing errors - See Sections 7.1.1.1.1 and 8.1.1.1). Finally, note that Equations (3.2.2.2-18) and (3.2.2.2-19) require a division by $1 - \cos\phi$ which is singular for $\cos\phi = 1$. This is the reason that Equations (3.2.2.2-18) and (3.2.2.2-19) are restricted to use when $\cos\phi < 0$ (i.e., $\pi / 2 < \phi \leq \pi$).

3.2.3 EULER ANGLES

A classical method for describing the attitude between two coordinate frames is through an Euler angle rotation sequence. An Euler angle sequence is a set of sequential rotations of a given coordinate frame around the frame's coordinate axes that positions it at a new attitude after the rotation sequence is completed. The final attitude of the displaced coordinate frame depends on the magnitude and axis of each of the sequential rotations in the selected Euler sequence. A common Euler angle sequence used to describe the attitude of "aircraft axes" relative to a locally level coordinate frame (with Z-axis down) consists of a "heading" rotation about the Z local level coordinate frame axis, followed by a "pitch" rotation about the displaced Y-axis, followed by a "roll" rotation about the displaced X-axis. The Euler angle sequence: heading (about Z), pitch (about Y), roll (about X), uniquely defines the attitude of the aircraft coordinate axes relative to the locally level coordinate frame.

To describe an Euler angle sequence mathematically, we can use the rotation vector concept described in Section 3.2.2 as applied to each of the selected Euler angle rotations in the selected sequence. With this approach, let's develop the analytics for the example aircraft axis Euler angle sequence discussed in the previous paragraph. First, we define coordinate frames for the selected Euler sequence where:

Frame A = Initial locally level coordinate frame.

Frame A_1 = Frame A after rotating it about axis Z through the heading Euler angle.

Frame A_2 = Frame A_1 after rotating it about axis Y through the pitch Euler angle.

Frame B = Aircraft axis frame obtained by rotating Frame A_2 about the X-axis through the roll Euler angle.

With the above coordinate frame definitions, we now define three rotation vectors (i.e., rotation axis $\underline{u}_\phi$ and angle ϕ as in Section 3.2.2) for each of the Euler angle rotations in the selected Euler sequence as follows:

$$\underline{u}_\phi = \underline{u}_{ZA} \qquad \phi = \psi$$
$$\underline{u}_\phi = \underline{u}_{YA_1} \qquad \phi = \theta \tag{3.2.3-1}$$
$$\underline{u}_\phi = \underline{u}_{XA_2} \qquad \phi = \phi$$

where

$\underline{u}_{ZA}, \underline{u}_{YA_1}, \underline{u}_{XA_2}$ = Unit vectors along the Frame A Z-axis, Frame A_1 Y-axis and Frame A_2 X-axis.

ψ, θ, ϕ = Amplitudes for each of the heading, pitch, roll Euler angle rotations (including sign). Note that ϕ is used here as the "roll" Euler angle (according to typical aircraft axis convention) and as the general rotation vector angle parameter, hopefully, without confusion.

We should note that in the definition of the rotation vector in Section 3.2.2, the general rotation vector angle parameter ϕ is a magnitude type quantity (i.e., plus by definition), with the rotation vector axis $\underline{u}_\phi$ reversing direction to account for negative rotations. In contrast, the general Euler rotation angle (ψ, θ or ϕ) can be plus or minus (i.e., it is not a magnitude type quantity), with the general Euler angle rotation axis ($\underline{u}_{ZA}$, $\underline{u}_{YA_1}$ or $\underline{u}_{XA_2}$) depending on the orientation of Frames A, A_1 and A_2. However, Equation (3.2.2.1-4) (direction cosines in terms of the rotation vector) was derived without explicitly requiring the rotation vector to be positive in sign. Hence, (3.2.2.1-4) is also valid if we allow the rotation angle to contain a sign (plus or minus). As such, Equation (3.2.2.1-4) can be applied to find the direction cosine matrices associated with the individual Euler angle rotations, treating the Euler rotation angles and axes as individual rotation vectors. Based on this finding, let us now calculate the three direction cosine matrices relating the previously defined coordinate frames using the (3.2.3-1) rotation vector definitions and Equation (3.2.2.1-4):

$$C_{A1}^{A} = I + \sin\psi\left(\underline{u}_{ZA}^{A}\times\right) + (1 - \cos\psi)\left(\underline{u}_{ZA}^{A}\times\right)\left(\underline{u}_{ZA}^{A}\times\right)$$
$$C_{A_2}^{A_1} = I + \sin\theta\left(\underline{u}_{YA_1}^{A_1}\times\right) + (1 - \cos\theta)\left(\underline{u}_{YA_1}^{A_1}\times\right)\left(\underline{u}_{YA_1}^{A_1}\times\right) \tag{3.2.3-2}$$
$$C_{B}^{A_2} = I + \sin\phi\left(\underline{u}_{XA_2}^{A_2}\times\right) + (1 - \cos\phi)\left(\underline{u}_{XA_2}^{A_2}\times\right)\left(\underline{u}_{XA_2}^{A_2}\times\right)$$

with, by their definition:

$$\underline{u}_{ZA}^{A} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \underline{u}_{YA_1}^{A_1} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \underline{u}_{XA_2}^{A_2} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{3.2.3-3}$$

Substituting Equations (3.2.3-3) into (3.2.3-2) obtains the more explicit direction cosine set:

$$C_{A_1}^{A} = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad C_{A_2}^{A_1} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$C_{B}^{A_2} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \tag{3.2.3-4}$$

3.2.3.1 DIRECTION COSINE MATRIX IN TERMS OF EULER ANGLE PARAMETERS

Equations (3.2.3-4) can be combined to yield the direction cosine matrix relating Frames A and B through successive application of the Equation (3.2.1-5) direction cosine matrix chain rule:

$$C_{B}^{A} = C_{A_1}^{A}\, C_{A_2}^{A_1}\, C_{B}^{A_2} \tag{3.2.3.1-1}$$

Substituting (3.2.3-4) into (3.2.3.1-1) obtains the relationship between the C_B^A direction cosine elements (as defined in Equations (3.2.1-1)) and the ψ, θ, ϕ Euler angles:

$$\begin{aligned}
C_{11} &= \cos\theta \cos\psi \\
C_{12} &= -\cos\phi \sin\psi + \sin\phi \sin\theta \cos\psi \\
C_{13} &= \sin\phi \sin\psi + \cos\phi \sin\theta \cos\psi \\[1ex]
C_{21} &= \cos\theta \sin\psi \\
C_{22} &= \cos\phi \cos\psi + \sin\phi \sin\theta \sin\psi \\
C_{23} &= -\sin\phi \cos\psi + \cos\phi \sin\theta \sin\psi \\[1ex]
C_{31} &= -\sin\theta \\
C_{32} &= \sin\phi \cos\theta \\
C_{33} &= \cos\phi \cos\theta
\end{aligned} \tag{3.2.3.1-2}$$

3.2.3.2 EULER ANGLES IN TERMS OF DIRECTION COSINES

Equations (3.2.3.1-2) can be inverted to obtain expressions for the Euler angle parameters in terms of the elements of the C_B^A direction cosine matrix. For the pitch angle parameter θ, the inverse solution is given by:

$$\theta = \tan^{-1} \frac{\sin \theta}{\cos \theta} = \tan^{-1} \frac{-C_{31}}{\sqrt{C_{32}^2 + C_{33}^2}} \tag{3.2.3.2-1}$$

Note that the positive square root solution has been selected for θ which bounds θ to never be greater than $\pi / 2$ in magnitude. This is the typical convention for the aircraft Euler angle sequence under study as an example. Note, however, that the negative square root solution would be equally valid producing three Euler angles that differ from the first set by the addition of π.

For the condition when $|\theta| \neq \pi / 2$ (or equivalently, from Equation (3.2.3.1-2), for $|C_{31}| \neq 1$), an inverse solution to Equations (3.2.3.1-2) for ψ and ϕ is obtainable as:

For $|C_{31}| \neq 1$ (e.g., for $|C_{31}| < 0.999$):

$$\phi = \tan^{-1} \frac{\sin \phi}{\cos \phi} = \tan^{-1} \frac{C_{32}}{C_{33}} \qquad \psi = \tan^{-1} \frac{\sin \psi}{\cos \psi} = \tan^{-1} \frac{C_{21}}{C_{11}} \tag{3.2.3.2-2}$$

When C_{31} approaches 1 (i.e., for $|\theta|$ approaching $\pi / 2$), Equations (3.2.3.2-2) become indeterminate because, from Equations (3.2.3.1-2), C_{11}, C_{21}, C_{32} and C_{33} all approach zero. For this situation, we can develop an inverse solution for the ψ and ϕ Euler angles, but only for their sum (or difference, depending on orientation). This solution is based on the following combined forms of Equations (3.2.3.1-2):

$$\begin{aligned} C_{23} - C_{12} &= (1 - C_{31}) \sin(\psi - \phi) \\ C_{13} + C_{22} &= (1 - C_{31}) \cos(\psi - \phi) \\ \\ C_{23} + C_{12} &= -(1 + C_{31}) \sin(\psi + \phi) \\ C_{13} - C_{22} &= -(1 + C_{31}) \cos(\psi + \phi) \end{aligned} \tag{3.2.3.2-3}$$

The top two expressions in (3.2.3.2-3) can be used to find $\psi - \phi$ when $1 - C_{31}$ is non-zero; the bottom two expressions can be used to find $\psi + \phi$ when $1 + C_{31}$ is non-zero. Thus, from Equations (3.2.3.2-3) we write when $\left| C_{31} \right| \geq 0.999$:

$$
\begin{aligned}
&\text{For } C_{31} \leq -0.999: \\
&\qquad \psi - \phi = \tan^{-1} \frac{C_{23} - C_{12}}{C_{13} + C_{22}} \\
&\text{For } C_{31} \geq 0.999: \\
&\qquad \psi + \phi = \pi + \tan^{-1} \frac{C_{23} + C_{12}}{C_{13} - C_{22}}
\end{aligned}
\tag{3.2.3.2-4}
$$

It is important to note that Equations (3.2.3.2-4) provide the only possible solution for the ψ and ϕ Euler angles under $\left| C_{31} \right|$ near unity conditions because when $\left| C_{31} \right|$ is near unity (i.e., when $\left| \theta \right|$ is near $\pi / 2$), the axes for the ψ and ϕ Euler rotations are collinear (i.e., the A Frame Z-axis and the A_2 Frame X-axis are parallel when $\left| \theta \right| = \pi / 2$). Under these conditions, ψ rotations about the A Frame Z-axis or ϕ rotations about the A_2 Frame X-axis have identical effects on the positioning of Frame B relative to Frame A. Consequently, only the sum (or difference) solutions as defined in Equations (3.2.3.2-4) can be obtained under these conditions.

3.2.3.3 METHOD OF LEAST WORK FOR TREATING EULER ROTATION OPERATIONS

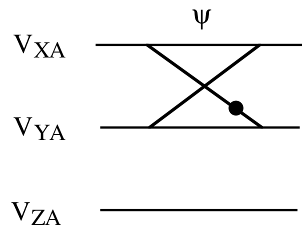
Signal flow theory has been used for the analysis of Euler angle sequences as a means for avoiding the complex matrix operations described in Section 3.2.3. To introduce this technique (affectionately denoted as "The Method Of Least Work" - attributable to Emery Curtis of Lockheed Missiles And Space Company in the early 1960's), we begin by the transformation of an arbitrary vector $\underline{V}$ from Frame A to Frame A_1 as defined in Section 3.2.3:

$$\underline{V}^{A1} = C_A^{A1} \, \underline{V}^A \tag{3.2.3.3-1}$$

With C_A^{A1} as the transpose of $C_{A_1}^A$ in Equations (3.2.3-4) and the vector notation convention introduced in Section 3.1, Equation (3.2.3.3-1) can be expanded in component form as follows:

$$
\begin{aligned}
V_{XA_1} &= V_{XA} \cos\psi + V_{YA} \sin\psi \\
V_{YA_1} &= V_{YA} \cos\psi - V_{XA} \sin\psi \\
V_{ZA_1} &= V_{ZA}
\end{aligned}
\tag{3.2.3.3-2}
$$

Equation (3.2.3.3-2) can be represented by the signal flow diagram in Figure 3.2.3.3-1:



$$V_{XA\,1} = V_{XA} \cos \psi + V_{YA} \sin \psi$$

$$V_{YA\,1} = V_{YA} \cos \psi - V_{XA} \sin \psi$$

$$V_{ZA\,1} = V_{ZA}$$

Figure 3.2.3.3-1 Z Axis Euler Rotation Signal Flow Diagram

which is interpreted as:

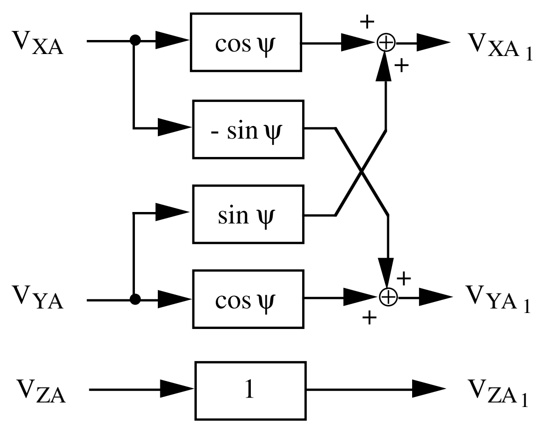
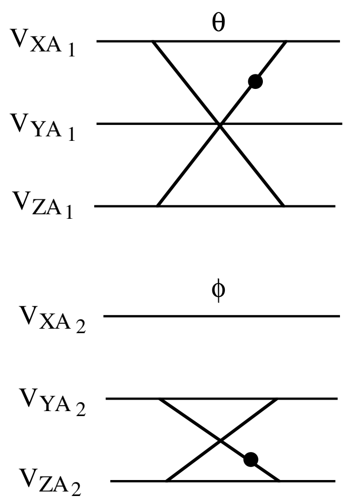


Figure 3.2.3.3-2 Interpretation Of Z Axis Euler Rotation Signal Flow Diagram

The horizontal lines in Figure 3.2.3.3-1 above and below the crossed lines are treated as transmission paths with a gain of cos ψ. The crossed lines in Figure 3.2.3.3-1 are treated as transmission paths with a gain of sin ψ. The dot (.) indicates minus (-) sin ψ. The dot is on the upward-left-to-downward-right (from left to right) diagonal if the ψ rotation is defined about the positive Z axis (as in Figure 3.2.3.3-1). For a rotation defined about the negative Z axis, the dot is on the downward-left-to-upward-right diagonal. The straight path alone in Figure 3.2.3.3-1 has unity gain. The $V_{XA\,1}$, $V_{YA\,1}$, $V_{ZA\,1}$ components are derived from the diagram by multiplying the V_{XA}, V_{YA}, V_{ZA} components on the left by the gains along all paths to the $V_{XA\,1}$, $V_{YA\,1}$, $V_{ZA\,1}$ components on the right, and summing the contributions to $V_{XA\,1}$, $V_{YA\,1}$ and $V_{ZA\,1}$. The result is Equations (3.2.3.3-2).

A similar derivation for the Frame A_1 Y axis (θ) rotation and Frame A_2 X-axis (ϕ) rotation yields:



$$V_{XA_2} = V_{XA_1} \cos\theta - V_{ZA_1} \sin\theta$$

$$V_{YA_2} = V_{YA_1}$$

$$V_{ZA_2} = V_{ZA_1} \cos\theta + V_{XA_1} \sin\theta$$

$$V_{XB} = V_{XA_2}$$

$$V_{YB} = V_{YA_2} \cos\phi + V_{ZA_2} \sin\phi$$

$$V_{ZB} = V_{ZA_2} \cos\phi - V_{YA_2} \sin\phi$$

Figure 3.2.3.3-3 Y And X Axis Euler Rotation Signal Flow Diagrams

The dots in Figure 3.2.3.3-3 are on the diagonals shown because the θ, ϕ rotations are defined about the positive Y, X axes. For θ, ϕ rotations defined about the negative Y, X axes, the dots would be on the other diagonals.

The heading, pitch, roll Euler sequence used for aircraft attitude referencing is then represented by the composite of Figures 3.2.3.3-1 and 3.2.3.3-3:

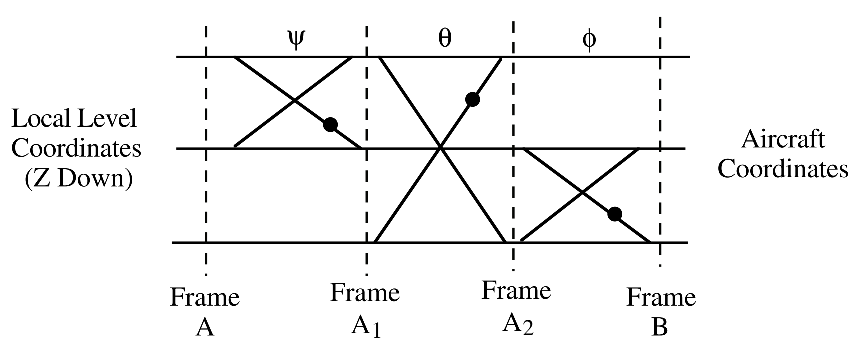


Figure 3.2.3.3-4 Aircraft Euler Angle Sequence Signal Flow Diagram

A vector $\underline{V}$ expressed in local level coordinates (Frame A) has equivalent components in aircraft coordinates (Frame B) equal to the Frame A component inputs at the left of Figure 3.2.3.3-4 multiplied by the net gain along each path to the right, with the results summed for each B Frame axis component. For example, for the YB component:

$$\begin{aligned} V_{YB} = \; & V_{XA} \left(\cos\psi \sin\theta \sin\phi - \sin\psi \cos\phi\right) \\ & + V_{YA} \left(\sin\psi \sin\theta \sin\phi + \cos\psi \cos\phi\right) + V_{ZA} \cos\theta \sin\phi \end{aligned} \tag{3.2.3.3-3}$$

A similar set can be obtained for the XB and ZB components of $\underline{V}$. It should be apparent that the terms in brackets in Equation (3.2.3.3-3) represent the cosines of the angles between Frame B coordinate axis Y and Frame A coordinate axes X, Y, and Z (i.e., the C_{I2} direction cosines in Equations (3.2.3.1-2)). The above procedure allows one to easily derive an analytical expression for the direction cosine between any left and right axis by tracing and summing all gains between the two points. The method is also reversible using the same diagram, for deriving analytical expressions for vectors on the left expressed in local level coordinates (Frame A) as a function of vector inputs on the right (in aircraft Frame B coordinates). Furthermore, a vector expressed in any of the intermediate frames (i.e., Frames A_1 or A_2) can be transformed to the A, B, or intermediate frames by inputting the components at the selected input frame station in the diagram and tracing the input signals to the desired output frame station. This is truly the method of least work for obtaining these expressions. Moreover, it is fun.

3.2.4 ATTITUDE REFERENCE QUATERNIONS AND QUATERNION COORDINATE FRAME TRANSFORMATIONS

The attitude reference quaternion is based on the rotation vector concept described in Section 3.2.2 that defines the attitude between two coordinate frames. The attitude quaternion associated with the two coordinate frames is defined as a set of four parameters: three of the parameters (defined as the vector part of the quaternion) equal the components of the rotation vector axis ($\underline{u}_\phi$) scaled by the sine of half the rotation vector angle ϕ; the fourth parameter is a scalar quantity equal to the cosine of half the rotation vector angle ϕ. To introduce the quaternion concept, let's begin the discussion in a somewhat unrelated field: complex numbers (as in Reference 25, Pages 73-76).

A complex number v is defined as having a real and imaginary part:

$$v = e + f\,i \tag{3.2.4-1}$$

where

e, f = Scalar quantities.

i = The imaginary number defined as the square root of minus one.

From the definition of i,

$$i\ i = -1 \tag{3.2.4-2}$$

The complex number v can be thought of as a "two-vector" with components e and f in the complex plane. Now consider another complex plane rotated from the first by an angle ψ (about an axis perpendicular to the real/imaginary axes). We will now demonstrate that another complex number u can be defined that can be used as an operator to transform the complex number v (treated as a vector) into its components along the rotated complex plane axes. Let's first define u in general as:

$$u = a + b\ i \tag{3.2.4-3}$$

where

a, b = Scalar quantities.

The product w of u with v is with (3.2.4-2):

$$\begin{aligned} w = u\ v = (a + b\ i)(e + f\ i) &= a\ e + a\ f\ i + b\ e\ i + b\ f\ i\ i \\ &= (e\ a - f\ b) + (f\ a + e\ b)\ i \end{aligned} \tag{3.2.4-4}$$

Hence, the effect of the multiplication operation of u on v is to create a new complex number w with a real component $(e\ a - f\ b)$ and an imaginary component $(f\ a + e\ b)$.

If the components of u are defined as

$$a = \cos\psi \qquad\qquad b = -\sin\psi \tag{3.2.4-5}$$

the u v product w in (3.2.4-4) would be:

$$w = (e\cos\psi + f\sin\psi) + (f\cos\psi - e\sin\psi)\ i \tag{3.2.4-6}$$

Comparing the form of (3.2.4-6) to the X and Y transformation operations in Equation (3.2.3.3-2), it should be apparent that the u v product vector represents vector v projected along the axes of a new complex plane rotated by ψ from the original. Thus, $u = \cos\psi - i\sin\psi$ can be considered as an operator that transforms vector v into a new coordinate frame rotated by ψ from the original frame.

Let's try to extend this concept into the world of three-dimensional vectors. If we now consider the i parameter to represent a unit vector along the X-axis of a three-dimensional coordinate frame, we can extend the concept of v to also include the Y and Z axis components as a "four-vector":

$$v = e + f\,i + g\,j + h\,k \qquad (3.2.4\text{-}7)$$

where

f, g, h = The conventional components of a vector in an X, Y, Z three-dimensional orthogonal coordinate frame.

i, j, k = Unit vectors along the coordinate frame axes.

e = A fourth component (a scalar) that would normally be zero if v represented a typical 3-component vector, but which is carried as a scalar quantity (e.g., in a fourth dimension) for the present.

The u quantity is similarly expanded.

$$u = a + b\,i + c\,j + d\,k \qquad (3.2.4\text{-}8)$$

where

b, c, d = Vector components of u.

a = Scalar component of u.

We now define the rules of four-vector multiplication by extension of the complex number concept using a right-handed vector cross-product convention:

$$\begin{array}{lll} i\,i = -1 & i\,j = k & i\,k = -j \\ j\,i = -k & j\,j = -1 & j\,k = i \\ k\,i = j & k\,j = -i & k\,k = -1 \end{array} \qquad (3.2.4\text{-}9)$$

With these definitions, the product w of u with v is given by:

$$w = u\,v = (a + b\,i + c\,j + d\,k)(e + f\,i + g\,j + h\,k)$$

$$\begin{aligned} &= a\,e + a\,f\,i + a\,g\,j + a\,h\,k && = a\,e - b\,f - c\,g - d\,h \\ &\quad + b\,e\,i + b\,f\,i\,i + b\,g\,i\,j + b\,h\,i\,k && \quad + (b\,e + a\,f - d\,g + c\,h)\,i \\ &\quad + c\,e\,j + c\,f\,j\,i + c\,g\,j\,j + c\,h\,j\,k && \quad + (c\,e + d\,f + a\,g - b\,h)\,j \\ &\quad + d\,e\,k + d\,f\,k\,i + d\,g\,k\,j + d\,h\,k\,k && \quad + (d\,e - c\,f + b\,g + a\,h)\,k \end{aligned} \qquad (3.2.4\text{-}10)$$

or in "four-vector" matrix form:

$$w = u\,v = \begin{bmatrix} e' \\ f' \\ g' \\ h' \end{bmatrix} = \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix} \qquad (3.2.4\text{-}11)$$

with

$$w \equiv e' + f'\,i + g'\,j + h'\,k \tag{3.2.4-12}$$

To complete the analogy it would be ideal at this point if we could now equate the components of u to a three-dimensional vector transformation operation and demonstrate that the i, j, k components of w as defined above represent the transformed version of the i, j, k components of v. Unfortunately, the analogy breaks down to a certain extent and such a simple relationship for u is not quite possible. However, an equivalent expression for u can be found that does possess the desired vector transformation property, if we modify the u operation on v to be defined as:

$$w = u\,v\,u^* \tag{3.2.4-13}$$

with

$$u^* = a - b\,i - c\,j - d\,k \tag{3.2.4-14}$$

where

u^* = Quaternion conjugate of u.

Carrying out the v u* product in Equation (3.2.4-13) using the previously stated rules of four-vector multiplication yields:

$$v\,u^* = (e + f\,i + g\,j + h\,k)(a - b\,i - c\,j - d\,k)$$

$$\begin{aligned} &= a\,e + b\,f + c\,g + d\,h \\ &\quad + (-\,b\,e + a\,f - d\,g + c\,h)\,i \\ &\quad + (-\,c\,e + d\,f + a\,g - b\,h)\,j \\ &\quad + (-\,d\,e - c\,f + b\,g + a\,h)\,k \end{aligned} = \begin{bmatrix} a & b & c & d \\ -b & a & -d & c \\ -c & d & a & -b \\ -d & -c & b & a \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix} \tag{3.2.4-15}$$

and for the newly defined w given by (3.2.4-13), we find by using v u* from (3.2.4-15) in place of v in (3.2.4-11):

$$w = u\,v\,u^* = \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix} \begin{bmatrix} a & b & c & d \\ -b & a & -d & c \\ -c & d & a & -b \\ -d & -c & b & a \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}$$

$$= \begin{bmatrix} (a^2 + b^2 + c^2 + d^2) & 0 & 0 & 0 \\ 0 & (a^2 + b^2 - c^2 - d^2) & 2\,(b\,c - a\,d) & 2\,(b\,d + a\,c) \\ 0 & 2\,(b\,c + a\,d) & (a^2 - b^2 + c^2 - d^2) & 2\,(c\,d - a\,b) \\ 0 & 2\,(b\,d - a\,c) & 2\,(c\,d + a\,b) & (a^2 - b^2 - c^2 + d^2) \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix} \tag{3.2.4-16}$$

We now equate the components of u to the previously defined rotation vector parameters. If the rotation vector angle for the rotation operation is ϕ and the rotation axis unit vector $\underline{u}_\phi$ is denoted as having i, j, k components of l, m and n, then the four components of u in accordance with quaternion convention (defined in the first paragraph of this section) are given by:

$$a = \cos\frac{\phi}{2} \qquad b = l\sin\frac{\phi}{2} \qquad c = m\sin\frac{\phi}{2} \qquad d = n\sin\frac{\phi}{2} \tag{3.2.4-17}$$

From the $\underline{u}_\phi$ expression in Equation (3.2.2-1), we can also write for the components of $\underline{u}_\phi$:

$$l = \frac{\phi_x}{\phi} \qquad m = \frac{\phi_y}{\phi} \qquad n = \frac{\phi_z}{\phi} \tag{3.2.4-18}$$

where

ϕ_x, ϕ_y, ϕ_z = The components of $\underline{\phi}$.

Thus,

$$a = \cos\frac{\phi}{2} \qquad b = \frac{\phi_x}{\phi}\sin\frac{\phi}{2} \qquad c = \frac{\phi_y}{\phi}\sin\frac{\phi}{2} \qquad d = \frac{\phi_z}{\phi}\sin\frac{\phi}{2} \tag{3.2.4-19}$$

Substituting Equation (3.2.4-19) into (3.2.4-16) yields after application of appropriate trigonometric identities:

$$w = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 - \left(\phi_y^2+\phi_z^2\right)\frac{(1-\cos\phi)}{\phi^2} & -\frac{\phi_z}{\phi}\sin\phi+\phi_x\phi_y\frac{(1-\cos\phi)}{\phi^2} & \frac{\phi_y}{\phi}\sin\phi+\phi_x\phi_z\frac{(1-\cos\phi)}{\phi^2} \\ 0 & \frac{\phi_z}{\phi}\sin\phi+\phi_x\phi_y\frac{(1-\cos\phi)}{\phi^2} & 1 - \left(\phi_x^2+\phi_z^2\right)\frac{(1-\cos\phi)}{\phi^2} & -\frac{\phi_x}{\phi}\sin\phi+\phi_y\phi_z\frac{(1-\cos\phi)}{\phi^2} \\ 0 & -\frac{\phi_y}{\phi}\sin\phi+\phi_x\phi_z\frac{(1-\cos\phi)}{\phi^2} & \frac{\phi_x}{\phi}\sin\phi+\phi_y\phi_z\frac{(1-\cos\phi)}{\phi^2} & 1 - \left(\phi_x^2+\phi_y^2\right)\frac{(1-\cos\phi)}{\phi^2} \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}$$

(3.2.4-20)

The lower right 3×3 elements in the (3.2.4-20) square matrix are equivalent to

$$I + \frac{\sin\phi}{\phi}\left(\underline{\phi}\times\right) + \frac{1-\cos\phi}{\phi^2}\left(\underline{\phi}\times\right)\left(\underline{\phi}\times\right)$$

which is identical to Equation (3.2.2.1-8) for the direction cosine matrix between two coordinate frames that are rotated relative to one another by the rotation vector $\underline{\phi}$. It can be concluded that the quaternion operation defined by Equation (3.2.4-13) with (3.2.4-19) for the u components is equivalent to a vector transformation operation on the three vector components of v.

We also note that by equating the upper diagonal element in Equations (3.2.4-16) to the upper diagonal element in (3.2.4-20) (or directly from Equations (3.2.4-19)) that:

$$a^2 + b^2 + c^2 + d^2 = 1 \qquad (3.2.4\text{-}21)$$

which is the normality characteristic of the attitude reference quaternion.

It is useful to note, as is easily verified by component expansion and substitution in Equations (3.2.4-10), that for any arbitrary quaternion:

$$u\, u^* = a^2 + b^2 + c^2 + d^2 \qquad (3.2.4\text{-}22)$$

Additionally, as is easily demonstrated by component expansion and substitution in Equations (3.2.4-10), the following conjugate product rule applies for two arbitrary quaternions u and v:

$$(u\, v)^* = v^*\, u^* \qquad (3.2.4\text{-}23)$$

Finally, the following mixed scalar/vector forms of (3.2.4-7) and (3.2.4-8) are sometimes useful in quaternion analytical operations:

$$u = a + \underline{r} \qquad\qquad v = e + \underline{s} \qquad (3.2.4\text{-}24)$$

in which

$$\underline{r} = b\, i + c\, j + d\, k \qquad\qquad \underline{s} = f\, i + g\, j + h\, k \qquad (3.2.4\text{-}25)$$

where

$\underline{r}$, $\underline{s}$ = Vector parts of the u, v quaternions.

In Equations (3.2.4-24), the $\underline{r}$ and $\underline{s}$ terms are treated as normal three component vectors for typical three component vector operations (e.g., cross product and dot product), but which follow the quaternion product rules for quaternion type products. Thus, with Equation (3.2.4-10):

$$\underline{r}\, \underline{s} = (c\, h - d\, g)\, i + (d\, f - b\, h)\, j + (b\, g - c\, f)\, k - (b\, f + c\, g + d\, h) \qquad (3.2.4\text{-}26)$$

From Equations (3.1.1-5) and (3.1.1-6) we see that (3.2.4-26) is equivalently:

$$\underline{r}\, \underline{s} = \underline{r} \times \underline{s} - \underline{r} \cdot \underline{s} \qquad (3.2.4\text{-}27)$$

Applying (3.2.4-24) and (3.2.4-27) then provides the mixed scalar/vector form of the general quaternion product:

$$u\,v \;=\; \underline{r} \times \underline{s} - \underline{r} \cdot \underline{s} + a\,\underline{s} + e\,\underline{r} + a\,e \tag{3.2.4-28}$$

3.2.4.1 QUATERNION OPERATIONS FOR ATTITUDE REFERENCE AND VECTOR TRANSFORMATIONS

The previous discussion has introduced the concept of the quaternion and its relationship to the direction cosine matrix. Following the notation convention introduced in Section 3.1 for the direction cosine matrix, we can now define the attitude reference quaternion relating arbitrary coordinate Frame A to arbitrary coordinate Frame B as a four element column array:

$$q_B^A \;=\; \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \tag{3.2.4.1-1}$$

in which, as in Section 3.2.4, a is the scalar portion and b, c, d is the vector portion, with the normalization characteristic of (3.2.4-21) repeated below:

$$a^2 + b^2 + c^2 + d^2 \;=\; 1 \tag{3.2.4.1-2}$$

We also define a general quaternion vector form in both Frame A and Frame B coordinates as the arrays:

$$v^A \;\equiv\; \begin{bmatrix} 0 \\ V_{XA} \\ V_{YA} \\ V_{ZA} \end{bmatrix} = \begin{bmatrix} 0 \\ \underline{V}^A \end{bmatrix} \qquad v^B \;\equiv\; \begin{bmatrix} 0 \\ V_{XB} \\ V_{YB} \\ V_{ZB} \end{bmatrix} = \begin{bmatrix} 0 \\ \underline{V}^B \end{bmatrix} \tag{3.2.4.1-3}$$

where

v^A, v^B = Quaternion equivalents to column matrix vectors $\underline{V}^A$ and $\underline{V}^B$ in Section 3.2.1 (See Equation (3.2.1-1)).

With the above definitions we can now write the following quaternion relations in which it is understood that the quaternion product rules still apply as defined in Section 3.2.4. From Equation (3.2.4-13), the conclusion following (3.2.4-20), and the above definitions we have:

$$v^A \;=\; q_B^A \, v^B \, q_B^{A^*} \tag{3.2.4.1-4}$$

Defining another arbitrary coordinate Frame D, we can also write:

$$v^A = q_D^A \, v^D \, q_D^{A^*} \qquad (3.2.4.1\text{-}5)$$

and

$$v^B = q_D^B \, v^D \, q_D^{B^*} \qquad (3.2.4.1\text{-}6)$$

Substituting Equation (3.2.4.1-6) into (3.2.4.1-4) then yields:

$$v^A = q_B^A \, q_D^B \, v^D \, q_D^{B^*} \, q_B^{A^*} \qquad (3.2.4.1\text{-}7)$$

or with Equation (3.2.4-23):

$$v^A = q_B^A \, q_D^B \, v^D \left(q_B^A \, q_D^B\right)^* \qquad (3.2.4.1\text{-}8)$$

Equating (3.2.4.1-5) and (3.2.4.1-8) then yields the coordinate frame chain rule for attitude quaternions:

$$q_D^A = q_B^A \, q_D^B \qquad (3.2.4.1\text{-}9)$$

From the definition of the rotation vector between Frames A and B and the property that the rotation vector has equal components in Frames A and B (See Section 3.2.2.1), it should be obvious that the rotation vector defining Frame B relative to Frame A is the negative of the rotation vector defining Frame A relative to Frame B. Applying this logic to the q_B^A attitude quaternion, we find from Equations (3.2.4-8), (3.2.4-14), (3.2.4-17) and (3.2.4.1-1) that since q_A^B has the negative of the q_B^A rotation vector, and vise-versa:

$$q_A^B = q_B^{A^*} \qquad q_B^A = q_A^{B^*} \qquad (3.2.4.1\text{-}10)$$

Equations (3.2.4.1-10) are analogous to Equations (3.2.1-3) for the equivalent direction cosine matrices.

3.2.4.2 DIRECTION COSINE MATRIX IN TERMS OF ATTITUDE QUATERNION

The C_B^A direction cosine matrix corresponding to the equivalent q_B^A attitude quaternion is shown by Equations (3.2.4.1-1), (3.2.4.1-3), (3.2.4.1-4), (3.2.4-16), (3.2.4-20), and (3.2.2.1-8) to be given by:

$$C_B^A = \begin{bmatrix} (a^2 + b^2 - c^2 - d^2) & 2\,(b\,c - a\,d) & 2\,(b\,d + a\,c) \\ 2\,(b\,c + a\,d) & (a^2 - b^2 + c^2 - d^2) & 2\,(c\,d - a\,b) \\ 2\,(b\,d - a\,c) & 2\,(c\,d + a\,b) & (a^2 - b^2 - c^2 + d^2) \end{bmatrix} \qquad (3.2.4.2\text{-}1)$$

3.2.4.3 ATTITUDE QUATERNION IN TERMS OF DIRECTION COSINES

Definition of the attitude quaternion parameters (a, b, c, d) in terms of the equivalent direction cosine matrix elements entails an inversion of Equation (3.2.4.2-1) (as in Reference 36).

For convenience, we first define:

$$P_a \equiv 4\,a^2 \qquad P_b \equiv 4\,b^2 \qquad P_c \equiv 4\,c^2 \qquad P_d \equiv 4\,d^2 \qquad (3.2.4.3\text{-}1)$$

The trace (sum of the diagonal terms) of the C_B^A direction matrix is given (from Equations (3.2.1-1) and (3.2.4.2-1) with (3.2.4.1-2) and (3.2.4.3-1)) by:

$$\begin{aligned} T_r &= C_{11} + C_{22} + C_{33} = 3\,a^2 - b^2 - c^2 - d^2 \\ &= 4\,a^2 - a^2 - b^2 - c^2 - d^2 = 4\,a^2 - 1 = P_a - 1 \end{aligned} \qquad (3.2.4.3\text{-}2)$$

or

$$P_a = 1 + T_r \qquad (3.2.4.3\text{-}3)$$

where

$$T_r = \text{Trace of } C_B^A.$$

From Equations (3.2.1-1) and (3.2.4.2-1) with (3.2.4-21):

$$C_{11} = a^2 + b^2 - c^2 - d^2 = 2\,a^2 + 2\,b^2 - a^2 - b^2 - c^2 - d^2 = 2\,a^2 + 2\,b^2 - 1 \qquad (3.2.4.3\text{-}4)$$

or with (3.2.4.3-1) and (3.2.4.3-2),

$$2\,C_{11} = P_a + P_b - 2 = P_b + T_r - 1 \qquad (3.2.4.3\text{-}5)$$

Hence,

$$P_b = 1 + 2\,C_{11} - T_r \qquad (3.2.4.3\text{-}6)$$

Similarly, including (3.2.4.3-6) and (3.2.4.3-3):

$$P_a = 1 + T_r \qquad P_b = 1 + 2\,C_{11} - T_r$$
$$P_c = 1 + 2\,C_{22} - T_r \qquad P_d = 1 + 2\,C_{33} - T_r \qquad (3.2.4.3\text{-}7)$$

The off-diagonal elements of Equation (3.2.4.2-1) with (3.2.1-1) show that:

$$b\,c = \frac{1}{4}\left(C_{21} + C_{12}\right) \qquad a\,d = \frac{1}{4}\left(C_{21} - C_{12}\right)$$
$$b\,d = \frac{1}{4}\left(C_{13} + C_{31}\right) \qquad a\,c = \frac{1}{4}\left(C_{13} - C_{31}\right) \qquad (3.2.4.3\text{-}8)$$
$$c\,d = \frac{1}{4}\left(C_{32} + C_{23}\right) \qquad a\,b = \frac{1}{4}\left(C_{32} - C_{23}\right)$$

The quaternion elements are now obtained as a function of the direction cosines from Equations (3.2.4.3-8) by first solving for one of the quaternion elements from the maximum of Equations (3.2.4.3-7) with (3.2.4.3-1), and then using this maximum to solve for the remaining three elements from (3.2.4.3-8). Thus:

If $P_a = \max\,(P_a, P_b, P_c, P_d)$, then:

$$a = 0.5\,\sqrt{P_a} \qquad b = \frac{C_{32} - C_{23}}{4\,a} \qquad c = \frac{C_{13} - C_{31}}{4\,a} \qquad d = \frac{C_{21} - C_{12}}{4\,a}$$

If $P_b = \max\,(P_a, P_b, P_c, P_d)$, then:

$$b = 0.5\,\sqrt{P_b} \qquad c = \frac{C_{21} + C_{12}}{4\,b} \qquad d = \frac{C_{13} + C_{31}}{4\,b} \qquad a = \frac{C_{32} - C_{23}}{4\,b}$$

If $P_c = \max\,(P_a, P_b, P_c, P_d)$, then: (3.2.4.3-9)

$$c = 0.5\,\sqrt{P_c} \qquad d = \frac{C_{32} + C_{23}}{4\,c} \qquad a = \frac{C_{13} - C_{31}}{4\,c} \qquad b = \frac{C_{21} + C_{12}}{4\,c}$$

If $P_d = \max\,(P_a, P_b, P_c, P_d)$, then:

$$d = 0.5\,\sqrt{P_d} \qquad a = \frac{C_{21} - C_{12}}{4\,d} \qquad b = \frac{C_{13} + C_{31}}{4\,d} \qquad c = \frac{C_{32} + C_{23}}{4\,d}$$

If $a \leq 0$, then:

$$a = -\,a \qquad b = -\,b \qquad c = -\,c \qquad d = -\,d$$

The final step in Equations (3.2.4.3-9) is based on the following rationale. From Equation (3.2.4.2-1), all direction cosine matrix elements consist of products of quaternion elements. Therefore, if the sign of all quaternion elements is changed, the solution for the direction cosine matrix in Equation (3.2.4.2-1) will be unaffected. This is equivalent to selecting the negative square root solution for the maximum P_i in Equations (3.2.4.3-7) rather than the positive solution used in the initial setting of the quaternion elements in Equations (3.2.4.3-9). Thus, there are two valid solutions for the quaternion elements corresponding to a given direction cosine matrix. The solution selected in the final step of Equations (3.2.4.3-9) is based on setting the a element positive. From Equations (3.2.4-19), this is equivalent to setting cos $\phi/2$ positive which corresponds to selecting $\left|\phi/2\right| < \pi/2$ or $\left|\phi\right| < \pi$. The solution for negative a corresponds to $\phi = \phi - 2\pi$ which is the rotation vector solution when the rotation to the new attitude is in the opposite direction about the same unit vector.

3.2.4.4 ATTITUDE QUATERNION IN TERMS OF ROTATION VECTOR

The Equation (3.2.4.1-1) q_B^A attitude quaternion elements as a function of the equivalent rotation vector components are given by (3.2.4-19) repeated below:

$$a = \cos\frac{\phi}{2} \qquad b = \frac{\phi_x}{\phi}\sin\frac{\phi}{2} \qquad c = \frac{\phi_y}{\phi}\sin\frac{\phi}{2} \qquad d = \frac{\phi_z}{\phi}\sin\frac{\phi}{2} \qquad (3.2.4.4\text{-}1)$$

Using the (3.2.4-24) mixed scalar/vector notation, Equations (3.2.4.4-1) with (3.2.2-1) and (3.2.4.1-1) become the compressed form:

$$q_B^A = \cos\frac{\phi}{2} + \frac{\sin\phi/2}{\phi}\,\underline{\phi} \qquad (3.2.4.4\text{-}2)$$

3.2.4.5 ROTATION VECTOR IN TERMS OF ATTITUDE QUATERNION

The rotation vector corresponding to a given attitude quaternion is obtained as the inverse of Equations (3.2.4.4-1). To assure that the resulting rotation vector magnitude is less than π, we apply the last of Equations (3.2.4.3-9) to the quaternion elements:

If $a \leq 0$, then:

$$a = -a \qquad b = -b \qquad c = -c \qquad d = -d \qquad (3.2.4.5\text{-}1)$$

With Equations (3.2.2-1) for the magnitude of ϕ, we obtain from (3.2.4.4-1):

$$\cos\frac{\phi}{2} = a \qquad \sin\frac{\phi}{2} = \sqrt{b^2 + c^2 + d^2} \qquad (3.2.4.5\text{-}2)$$

Note that the positive solution has been selected for sin $\phi/2$ to assure a positive rotation vector magnitude. From Equations (3.2.4.5-2) we find:

$$\frac{\phi}{2} = \tan^{-1} \frac{\sin \phi/2}{\cos \phi/2} = \tan^{-1} \frac{\sqrt{b^2 + c^2 + d^2}}{a} \qquad (3.2.4.5\text{-}3)$$

We then calculate the ratio of sin $\phi/2$ to ϕ as a Taylor series expansion:

$$f \equiv \frac{\sin \phi/2}{\phi} = \frac{1}{2} \left(1 - \frac{(\phi/2)^2}{3\,!} + \frac{(\phi/2)^4}{5\,!} - \frac{(\phi/2)^6}{7\,!} + \cdots \right) \qquad (3.2.4.5\text{-}4)$$

Finally, f from Equation (3.2.4.5-4) is used in the last three of Equations (3.2.4.4-1) to yield after rearrangement:

$$\phi_x = b / f \qquad \phi_y = c / f \qquad \phi_z = d / f \qquad (3.2.4.5\text{-}5)$$

Note, that because of the Equation (3.2.4.5-1) check process which assures that cos $\phi/2$ is positive, $|\phi|$ is less than π, hence, $|\phi/2|$ is less than $\pi/2$. This assures that f from Equation (3.2.4.5-4) will never be zero, thereby avoiding a singularity condition for Equations (3.2.4.5-5).

3.3 ATTITUDE PARAMETER RATE EQUATIONS

Section 3.2 described the parameters that are typically incorporated to describe the angular attitude orientation between two coordinate frames (the direction cosine matrix, the rotation vector, Euler angles and the attitude quaternion). In this section we derive equations describing the rate of change of the Section 3.2 attitude parameters as a function of the angular rates of their defining coordinate frames. The fundamental basis for the attitude parameter rate equations is the general Coriolis relationship that describes how a constant vector in one coordinate frame appears in another coordinate frame that is rotating relative to the first.

3.3.1 GENERAL CORIOLIS RELATIONSHIP BETWEEN VECTORS IN ROTATING COORDINATE FRAMES

Consider an arbitrary vector $\underline{V}_{CnstB}$ that is fixed (constant) in an arbitrary coordinate Frame B. Now, assume that Frame B is rotating relative to arbitrary coordinate Frame A at angular velocity $\underline{\omega}_{AB}$. Define the $\underline{V}_{CnstB}$ and $\underline{\omega}_{AB}$ components in coordinate Frame A as the column vectors $\underline{V}^A_{CnstB}$ and $\underline{\omega}^A_{AB}$ where:

A, B = Arbitrary coordinate frames.

$\underline{V}_{CnstB}^{A}$ = Arbitrary vector that is constant in the B Frame (subscript) as projected on A Frame axes (superscript).

$\underline{\omega}_{AB}^{A}$ = Angular rate of Frame B relative to Frame A as projected on Frame A axes.

Further, assume that the angle between $\underline{V}_{CnstB}$ and $\underline{\omega}_{AB}$ is α. Figure 3.3.1-1 is a view of the situation from the perspective of Frame A.

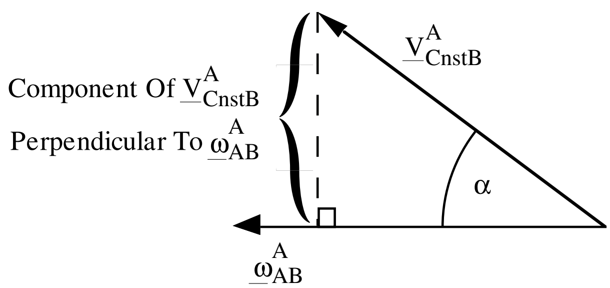


Figure 3.3.1-1 The Situation As Viewed In Frame A

From the viewpoint of Frame A with α less than π (as depicted in Figure 3.3.1-1), $\underline{V}_{CnstB}^{A}$ appears to be rotating around $\underline{\omega}_{AB}^{A}$, thereby producing a $\underline{V}_{CnstB}^{A}$ rate of change into the plane of the paper (i.e., perpendicular to $\underline{\omega}_{AB}^{A}$ and $\underline{V}_{CnstB}^{A}$) equal to the magnitude of the component of $\underline{V}_{CnstB}^{A}$ perpendicular to $\underline{\omega}_{AB}^{A}$ times the magnitude of $\underline{\omega}_{AB}^{A}$. The same is true for α greater than π (and less than 2 π), except that the $\underline{V}_{CnstB}^{A}$ rate of change direction would be out of the plane of the paper. From Figure 3.3.1-1, the magnitude of the component of $\underline{V}_{CnstB}^{A}$ perpendicular to $\underline{\omega}_{AB}^{A}$ equals the magnitude of $\underline{V}_{CnstB}^{A}$ multiplied by the magnitude of the sine of the angle between $\underline{\omega}_{AB}^{A}$ and $\underline{V}_{CnstB}^{A}$, hence:

$$\left|\dot{\underline{V}}_{CnstB}^{A}\right| = \left|\underline{\omega}_{AB}^{A}\right|\left|\underline{V}_{CnstB}^{A}\right|\left|\sin\alpha\right| \tag{3.3.1-1}$$

where

$\dot{\underline{V}}_{CnstB}^{A}$ = Time rate of change of $\underline{V}_{CnstB}^{A}$.

From the Equation (3.1.1-3) definition of the cross-product between two vectors (and the notes following (3.1.1-3)), the above magnitude and direction properties of $\dot{\underline{V}}^A_{CnstB}$ show that:

$$\dot{\underline{V}}^A_{CnstB} = \underline{\omega}^A_{AB} \times \underline{V}^A_{CnstB} \tag{3.3.1-2}$$

Equation (3.3.1-2) is a fundamental Coriolis relationship defining the rate of change in coordinate Frame A of the components of a vector $\underline{V}^A_{CnstB}$ that is fixed in coordinate Frame B, when coordinate Frame B is rotating relative to coordinate Frame A at angular velocity $\underline{\omega}^A_{AB}$. Since the A and B Frames are arbitrary, Equation (3.3.1-2) is equally valid if we interchange A and B:

$$\dot{\underline{V}}^B_{CnstA} = \underline{\omega}^B_{BA} \times \underline{V}^B_{CnstA} \tag{3.3.1-3}$$

where

$\underline{\omega}^B_{BA}$ = Angular rate of Frame A relative to Frame B as projected on Frame B axes.

$\underline{V}^B_{CnstA}$ = Arbitrary vector that is constant in the A Frame (subscript) as projected on B Frame axes (superscript).

We also note that the angular rate of Frame A relative to Frame B is the negative of the angular rate of Frame B relative to Frame A so that:

$$\underline{\omega}^B_{BA} = - \underline{\omega}^B_{AB} \tag{3.3.1-4}$$

where

$\underline{\omega}^B_{AB}$ = Angular rate of Frame B relative to Frame A as projected on Frame B axes.

With (3.3.1-4), Equation (3.3.1-3) is equivalently:

$$\dot{\underline{V}}^B_{CnstA} = - \underline{\omega}^B_{AB} \times \underline{V}^B_{CnstA} \tag{3.3.1-5}$$

Equation (3.3.1-5), the complement to Coriolis Equation (3.3.1-2), defines the rate of change in coordinate Frame B of the components a vector $\underline{V}^B_{CnstA}$ that is fixed in coordinate Frame A, when coordinate Frame B is rotating relative to coordinate Frame A at angular velocity $\underline{\omega}^B_{AB}$.

3.3.2 DIRECTION COSINE MATRIX RATE EQUATION

Equation (3.3.1-5) can be used to derive an equation for the rate of change of the direction cosine matrix C_B^A relating general coordinate Frames A and B. As defined in Equations (3.2.1-6), the columns of C_A^B (i.e., the transpose of C_B^A) represent unit vectors along Frame A coordinate axes as projected onto Frame B. Therefore, C_B^A is given by:

$$C_B^A = \left(\underline{u}_{1A}^B, \underline{u}_{2A}^B, \underline{u}_{3A}^B\right)^T \tag{3.3.2-1}$$

where

$\underline{u}_{JA}^B$ = The column vector whose elements represent the Frame B components of a unit vector along the J^{th} Frame A coordinate axis.

Taking the derivative of (3.3.2-1) obtains:

$$\dot{C}_B^A = \left(\dot{\underline{u}}_{1A}^B, \dot{\underline{u}}_{2A}^B, \dot{\underline{u}}_{3A}^B\right)^T \tag{3.3.2-2}$$

Applying Equation (3.3.1-5):

$$\dot{\underline{u}}_{JA}^B = - \underline{\omega}_{AB}^B \times \underline{u}_{JA}^B \tag{3.3.2-3}$$

where

$\underline{\omega}_{AB}^B$ = Angular rate of Frame B relative to Frame A (as projected on Frame B axes).

or

$$\dot{\underline{u}}_{JA}^B = - \left(\underline{\omega}_{AB}^B \times\right) \underline{u}_{JA}^B \tag{3.3.2-4}$$

where

$$\left(\underline{\omega}_{AB}^B \times\right) = \text{Skew symmetric form of } \underline{\omega}_{AB}^B \equiv \begin{bmatrix} 0 & -\omega_{AB_{ZB}} & \omega_{AB_{YB}} \\ \omega_{AB_{ZB}} & 0 & -\omega_{AB_{XB}} \\ -\omega_{AB_{YB}} & \omega_{AB_{XB}} & 0 \end{bmatrix}$$

$\omega_{AB_{XB}}, \omega_{AB_{YB}}, \omega_{AB_{ZB}}$ = X, Y, Z components of $\underline{\omega}_{AB}^B$.

Substituting Equation (3.3.2-4) into (3.3.2-2) then yields:

$$\dot{C}_B^A = -\left[\left(\underline{\omega}_{AB}^B \times\right)\left(\underline{u}_{1A}^B, \underline{u}_{2A}^B, \underline{u}_{3A}^B\right)\right]^T = -\left(\underline{u}_{1A}^B, \underline{u}_{2A}^B, \underline{u}_{3A}^B\right)^T \left(\underline{\omega}_{AB}^B \times\right)^T$$
$$= \left(\underline{u}_{1A}^B, \underline{u}_{2A}^B, \underline{u}_{3A}^B\right)^T \left(\underline{\omega}_{AB}^B \times\right) \qquad (3.3.2\text{-}5)$$

in which use has been made of the fact that the transpose of a skew symmetric matrix equals the negative of the matrix. With Equation (3.3.2-1), we finally obtain:

$$\dot{C}_B^A = C_B^A \left(\underline{\omega}_{AB}^B \times\right) \qquad (3.3.2\text{-}6)$$

The rate of change of C_B^A can also be defined in a more familiar form in terms of the individual inertial angular rotation rates of Frames A and B (i.e., angular rotation rates relative to non-rotating inertial space). Using the Equation (3.2.1-5) matrix product chain rule, we first define C_B^A as:

$$C_B^A = C_I^A \, C_B^I \qquad (3.3.2\text{-}7)$$

where

I = Non-rotating inertial coordinate frame.

C_I^A, C_B^I = Direction cosine matrices relating the inertial non-rotating coordinate Frame I to Frame A and Frame B.

The derivative of (3.3.2-7) is:

$$\dot{C}_B^A = C_I^A \, \dot{C}_B^I + \dot{C}_I^A \, C_B^I \qquad (3.3.2\text{-}8)$$

Using the same procedure leading to Equation (3.3.2-6) allows us to write:

$$\dot{C}_B^I = C_B^I \left(\underline{\omega}_{IB}^B \times\right) \qquad (3.3.2\text{-}9)$$

and

$$\dot{C}_A^I = C_A^I \left(\underline{\omega}_{IA}^A \times\right) \qquad (3.3.2\text{-}10)$$

where

$\underline{\omega}_{IB}^B$ = Angular rate of Frame B relative to inertial Frame I (as projected on Frame B axes).

$\underline{\omega}_{IA}^A$ = Angular rate of Frame A relative to inertial Frame I (as projected on Frame A axes).

$\left(\underline{\omega}_{IA}^{A}\times\right)$ = Skew symmetric matrix form of $\underline{\omega}_{IA}^{A}$.

$\left(\underline{\omega}_{IB}^{B}\times\right)$ = Skew symmetric matrix form of $\underline{\omega}_{IB}^{B}$.

Taking the transpose of (3.3.2-10) obtains:

$$\dot{C}_{I}^{A} = -\left(\underline{\omega}_{IA}^{A}\times\right) C_{I}^{A} \tag{3.3.2-11}$$

Substituting Equations (3.3.2-9) and (3.3.2-11) into (3.3.2-8) then yields:

$$\dot{C}_{B}^{A} = C_{I}^{A}\left[C_{B}^{I}\left(\underline{\omega}_{IB}^{B}\times\right)\right] - \left[\left(\underline{\omega}_{IA}^{A}\times\right) C_{I}^{A}\right] C_{B}^{I} \tag{3.3.2-12}$$

or, upon recombining the matrix elements through Equation (3.3.2-7):

$$\dot{C}_{B}^{A} = C_{B}^{A}\left(\underline{\omega}_{IB}^{B}\times\right) - \left(\underline{\omega}_{IA}^{A}\times\right) C_{B}^{A} \tag{3.3.2-13}$$

Equation (3.3.2-13) relates the rate of change of the Frame B to Frame A direction cosine matrix to the inertial rotation rate of Frame B as defined in Frame B axes and the inertial rotation rate of Frame A as defined in Frame A axes. An important characteristic of Equation (3.3.2-13) is that the coefficient matrix multiplying the angular rates is a direction cosine matrix, hence, its components can never exceed one in magnitude. Therefore, assuming that the A and B Frames are chosen to have finite angular rates, Equation (3.3.2-13) has no singularities for any attitude of Frame B relative to Frame A.

It is instructive to also examine the component form of Equation (3.3.2-13). Defining the matrix elements as:

$$\underline{\omega}_{IB}^{B} \equiv \begin{bmatrix} \omega_{XB} \\ \omega_{YB} \\ \omega_{ZB} \end{bmatrix} \qquad \underline{\omega}_{IA}^{A} \equiv \begin{bmatrix} \omega_{XA} \\ \omega_{YA} \\ \omega_{ZA} \end{bmatrix} \qquad C_{B}^{A} \equiv \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \tag{3.3.2-14}$$

we have:

$$
\begin{aligned}
\dot{C}_{11} &= C_{12}\,\omega_{ZB} - C_{13}\,\omega_{YB} + C_{21}\,\omega_{ZA} - C_{31}\,\omega_{YA} \\
\dot{C}_{12} &= C_{13}\,\omega_{XB} - C_{11}\,\omega_{ZB} + C_{22}\,\omega_{ZA} - C_{32}\,\omega_{YA} \\
\dot{C}_{13} &= C_{11}\,\omega_{YB} - C_{12}\,\omega_{XB} + C_{23}\,\omega_{ZA} - C_{33}\,\omega_{YA} \\
\\
\dot{C}_{21} &= C_{22}\,\omega_{ZB} - C_{23}\,\omega_{YB} + C_{31}\,\omega_{XA} - C_{11}\,\omega_{ZA} \\
\dot{C}_{22} &= C_{23}\,\omega_{XB} - C_{21}\,\omega_{ZB} + C_{32}\,\omega_{XA} - C_{12}\,\omega_{ZA} \\
\dot{C}_{23} &= C_{21}\,\omega_{YB} - C_{22}\,\omega_{XB} + C_{33}\,\omega_{XA} - C_{13}\,\omega_{ZA} \\
\\
\dot{C}_{31} &= C_{32}\,\omega_{ZB} - C_{33}\,\omega_{YB} + C_{11}\,\omega_{YA} - C_{21}\,\omega_{XA} \\
\dot{C}_{32} &= C_{33}\,\omega_{XB} - C_{31}\,\omega_{ZB} + C_{12}\,\omega_{YA} - C_{22}\,\omega_{XA} \\
\dot{C}_{33} &= C_{31}\,\omega_{YB} - C_{32}\,\omega_{XB} + C_{13}\,\omega_{YA} - C_{23}\,\omega_{XA}
\end{aligned}
\qquad (3.3.2\text{-}15)
$$

3.3.3 EULER ANGLE RATE EQUATIONS

An expression for the rate of change of the Euler angles in an Euler sequence can be developed as a function of the relative angular rate between the coordinate frames connected by the Euler sequence. The method is to recognize that the total angular rate vector between the two coordinate frames is comprised of the vector sum of the individual Euler angle rates treated as vectors along their respective rotation axes. Hence, for the particular aircraft Euler angle sequence treated in Section 3.2.3 that defines the attitude of arbitrary coordinate Frames A and B, we can write:

$$\underline{\omega}_{AB} = \dot{\psi}\,\underline{u}_{ZA} + \dot{\theta}\,\underline{u}_{YA_1} + \dot{\phi}\,\underline{u}_{XA_2} \qquad (3.3.3\text{-}1)$$

in which the contributing terms are as defined in Section 3.2.3.

As in Section 3.3.2, it is convenient to define $\underline{\omega}_{AB}$ as the difference between the individual Frame A and Frame B angular rotation rates relative to an inertial non-rotating coordinate Frame I:

$$\underline{\omega}_{AB} = \underline{\omega}_{IB} - \underline{\omega}_{IA} \qquad (3.3.3\text{-}2)$$

Substituting Equation (3.3.3-2) into (3.3.3-1) and projecting the result on Frame B axes using the (3.2.1-5) chain rule then yields:

$$\underline{\omega}_{IB}^{B} - C_{A}^{B}\,\underline{\omega}_{IA}^{A} = \dot{\psi}\,C_{A_2}^{B}\,C_{A_1}^{A_2}\,\underline{u}_{ZA}^{A_1} + \dot{\theta}\,C_{A_2}^{B}\,\underline{u}_{YA_1}^{A_2} + \dot{\phi}\,\underline{u}_{XA_2}^{B} \qquad (3.3.3\text{-}3)$$

3.3.3.1 FRAME B ROTATION RATES IN TERMS OF EULER RATES AND FRAME A RATES

Each unit vector for each Euler rotation in the Euler sequence (See Section 3.2.3) has identical components in the coordinate frame before and after the Euler rotation about the unit vector, hence:

$$\underline{u}_{ZA}^{A_1} = \underline{u}_{ZA}^{A} \qquad \underline{u}_{YA_1}^{A_2} = \underline{u}_{YA_1}^{A_1} \qquad \underline{u}_{XA_2}^{B} = \underline{u}_{XA_2}^{A_2} \tag{3.3.3.1-1}$$

We can also write from (3.2.1-3):

$$C_A^B = \left(C_B^A\right)^T \qquad C_{A_2}^B = \left(C_B^{A_2}\right)^T \qquad C_{A_1}^{A_2} = \left(C_{A_2}^{A_1}\right)^T$$

$$\underline{\omega}_{IB}^{B} \equiv \begin{bmatrix} \omega_{XB} \\ \omega_{YB} \\ \omega_{ZB} \end{bmatrix} \qquad \underline{\omega}_{IA}^{A} \equiv \begin{bmatrix} \omega_{XA} \\ \omega_{YA} \\ \omega_{ZA} \end{bmatrix} \tag{3.3.3.1-2}$$

Substituting Equations (3.3.3.1-1) and (3.3.3.1-2) into (3.3.3-3) with the Euler angle direction cosine matrix component forms provided by Equations (3.2.3-4), yields expressions for the Frame B angular rate components in terms of the Euler angle rates and Frame A angular rates:

$$\begin{aligned}
\omega_{XB} &= \dot{\phi} - \dot{\psi} \sin\theta + \omega_{XA} \cos\theta \cos\psi + \omega_{YA} \cos\theta \sin\psi - \omega_{ZA} \sin\theta \\
\omega_{YB} &= \dot{\theta} \cos\phi + \dot{\psi} \cos\theta \sin\phi + \omega_{XA} \left(- \cos\phi \sin\psi + \sin\phi \sin\theta \cos\psi\right) \\
&\quad + \omega_{YA} \left(\cos\phi \cos\psi + \sin\phi \sin\theta \sin\psi\right) + \omega_{ZA} \sin\phi \cos\theta \\
\omega_{ZB} &= - \dot{\theta} \sin\phi + \dot{\psi} \cos\theta \cos\phi + \omega_{XA} \left(\sin\phi \sin\psi + \cos\phi \sin\theta \cos\psi\right) \\
&\quad + \omega_{YA} \left(- \sin\phi \cos\psi + \cos\phi \sin\theta \sin\psi\right) + \omega_{ZA} \cos\phi \cos\theta
\end{aligned} \tag{3.3.3.1-3}$$

An important characteristic of Equations (3.3.3.1-3) is that the coefficients multiplying the angular rates are unity or products of sines and cosines, hence, the coefficients can never exceed one in magnitude. Therefore, assuming that the Euler angle rates and Frame A rates are chosen to be finite, Equation (3.3.3.1-3) has no singularities for any attitude of Frame B relative to Frame A.

3.3.3.2 EULER ANGLE RATES IN TERMS OF FRAME B AND FRAME A RATES

Successive applications of the Equation (3.2.1-5) chain rule shows that:

$$C_A^B = C_{A_2}^B \, C_{A_1}^{A_2} \, C_A^{A_1} \tag{3.3.3.2-1}$$

Multiplying Equation (3.3.3-3) by $C_B^{A_2}$ yields after applying (3.3.3.2-1):

$$C_B^{A_2} \, \underline{\omega}_{IB}^B - C_{A_1}^{A_2} \, C_A^{A_1} \, \underline{\omega}_{IA}^A = \dot{\psi} \, C_{A_1}^{A_2} \, \underline{u}_{ZA}^{A_1} + \dot{\theta} \, \underline{u}_{YA_1}^{A_2} + \dot{\phi} \, C_B^{A_2} \, \underline{u}_{XA_2}^B \tag{3.3.3.2-2}$$

Substituting Equations (3.3.3.1-1) and (3.3.3.1-2) into (3.3.3.2-2) with the Euler angle direction cosine matrix component forms provided by Equations (3.2.3-4) yields expressions for the Frame A and B angular rate components in terms of the Euler angle rates:

$$\begin{aligned}
&\omega_{XB} - \omega_{XA} \cos\theta \cos\psi - \omega_{YA} \cos\theta \sin\psi + \omega_{ZA} \sin\theta = -\dot{\psi} \sin\theta + \dot{\phi} \\
&\omega_{YB} \cos\phi - \omega_{ZB} \sin\phi + \omega_{XA} \sin\psi - \omega_{YA} \cos\psi = \dot{\theta} \\
&\omega_{YB} \sin\phi + \omega_{ZB} \cos\phi - \omega_{XA} \sin\theta \cos\psi - \omega_{YA} \sin\theta \sin\psi - \omega_{ZA} \cos\theta = \dot{\psi} \cos\theta
\end{aligned} \tag{3.3.3.2-3}$$

Rearrangement and combination of Equations (3.3.3.2-3) then obtains the desired expressions for the Euler angle rates in terms of the Frame A and B angular rates:

$$\begin{aligned}
&\dot{\phi} = \omega_{XB} + \omega_{YB} \tan\theta \sin\phi + \omega_{ZB} \tan\theta \cos\phi - \omega_{XA} \sec\theta \cos\psi - \omega_{YA} \sec\theta \sin\psi \\
&\dot{\theta} = \omega_{YB} \cos\phi - \omega_{ZB} \sin\phi + \omega_{XA} \sin\psi - \omega_{YA} \cos\psi \\
&\dot{\psi} = \omega_{YB} \sec\theta \sin\phi + \omega_{ZB} \sec\theta \cos\phi - \omega_{XA} \tan\theta \cos\psi - \omega_{YA} \tan\theta \sin\psi - \omega_{ZA}
\end{aligned} \tag{3.3.3.2-4}$$

An important characteristic of Equations (3.3.3.2-4) is that the $\dot{\phi}$ and $\dot{\psi}$ expressions contain singularities due to $\sec\theta$ and $\tan\theta$ terms that are infinite at $|\theta| = \pi/2$. This is a manifestation of the Euler angle singularity situation discussed in Section 3.2.3.2.

3.3.3.3 METHOD OF LEAST WORK FOR EULER RATE EQUATION DERIVATION

An interesting application of the Section 3.2.3.3 "Method Of Least Work" technique is determination of the $\underline{\omega}_{IB}^B$ components from Euler angle rates (normally found through laborious standard matrix algebraic and component conversion techniques). By introducing each of the Equation (3.3.3-1) $\dot{\phi}, \dot{\theta}, \dot{\psi}$ vectors into the Section 3.2.3.3 Euler angle sequence signal flow diagram at the node points for which their rotation axis is defined, and then tracing and summing to the right, the components of $\underline{\omega}_{IB}^{B}$ are determined. The $\dot{\phi}, \dot{\theta}, \dot{\psi}$ quantities are around X, Y, Z respectively in the intermediate frames in which their Euler angles are defined (i.e., from Equation (3.3.3-1), $\dot{\psi}$ around $\underline{u}_{ZA}$, $\dot{\theta}$ around $\underline{u}_{YA_1}$, and $\dot{\phi}$ around $\underline{u}_{XA_2}$). Thus:

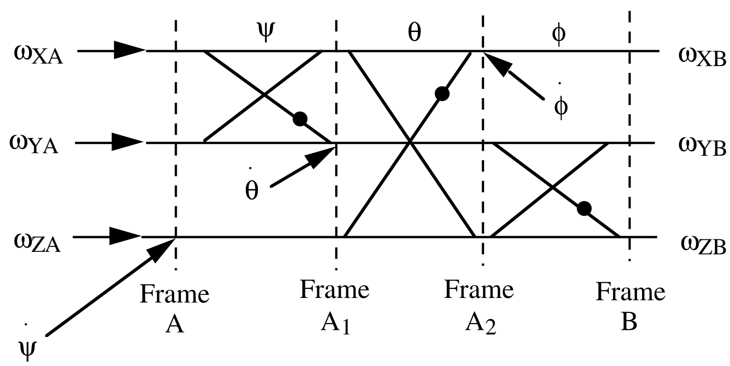


Figure 3.3.3.3-1 B Frame Rates From Euler Angle Rates Using The Method Of Least Work

The Figure 3.3.3.3-1 diagram is based on the Euler rotations being about positive X, Y, Z axes. If any of the rotations are defined to be about negative axes (X, Y or Z), the associated diagonal dot would be switched to the other diagonal (see explanation following Figures 3.2.3.3-2 and 3.2.3.3-3), and the associated Euler angle rate input would have a negative sign.

Using Figure 3.3.3.3-1, each of Equations (3.3.3.1-3) can be readily obtained. What could be simpler?

The diagram also works in the inverse direction, provided that the three outputs are calculated at one coordinate frame location. Thus, each of Equations (3.3.3.2-3) can be obtained from outputs taken at the Frame A_2 nodes. Unfortunately, obtaining $\dot{\phi}, \dot{\theta}, \dot{\psi}$ directly from the diagram in terms of the Frame A and B angular rates (i.e., as in Equations (3.3.3.2-4)) is not as easily achieved without some trickery (left as an exercise).

3.3.4 ATTITUDE QUATERNION RATE EQUATION

An equation for the rate of change of the q_B^A attitude quaternion can be derived in terms of the relative angular rate between coordinate Frame A and B by envisioning the Frame B rotation rate, from Frame B's standpoint, as relative to a stationary Frame A. We consider Frame B to be rotating from orientation 1 to orientation 2, assuming the 1, 2 orientations to be angularly (and time-wise) very close to one another. Applying these considerations in the Equation (3.2.4.1-9) quaternion chain rule:

$$q_{B_2}^A = q_{B_1}^A \, q_{B_2}^{B_1} \tag{3.3.4-1}$$

where

B_1 = Instantaneous orientation of Frame B at some arbitrary time, considered stationary in the A Frame (i.e., a "benchmark" attitude orientation).

B_2 = New orientation of Frame B following its previous Frame B_1 attitude, considering B_2 to be close to B_1 (in time and attitude).

The change in q_B^A from the B_1 to B_2 orientations is:

$$\Delta q_B^A = q_{B_2}^A - q_{B_1}^A \tag{3.3.4-2}$$

where

Δq_B^A = Change in q_B^A as Frame B rotates from attitude B_1 to attitude B_2.

or, with (3.3.4-1) while recognizing that B_1 and B_2 are close to one another at the general Frame B attitude:

$$\Delta q_B^A = q_{B_1}^A \, q_{B_2}^{B_1} - q_{B_1}^A = q_{B_1}^A \left(q_{B_2}^{B_1} - q_1 \right) \approx q_B^A \left(q_{B_2}^{B_1} - q_1 \right) \tag{3.3.4-3}$$

with

$$q_1 \equiv \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.3.4-4}$$

where

q_1 = Identity attitude quaternion analogous to the identity matrix for a direction cosine matrix relating two coincident coordinate frames.

We now define a rotation vector $\underline{\phi}$ associated with $q_{B_2}^{B_1}$. Since B_2 and B_1 are near each other angularly, $\underline{\phi}$ will be small in magnitude, allowing Equation (3.2.4.4-2) for $q_{B_2}^{B_1}$ to be approximated by:

$$q_{B_2}^{B_1} \approx \begin{bmatrix} 1 \\ \frac{1}{2}\underline{\phi} \end{bmatrix} \tag{3.3.4-5}$$

so that Equation (3.3.4-3) becomes:

$$\Delta q_B^A \approx \frac{1}{2} q_B^A \begin{bmatrix} 0 \\ \underline{\phi} \end{bmatrix} \tag{3.3.4-6}$$

Dividing Equation (3.3.4-6) by the small time interval for movement of B from B_1 to B_2 and letting the time interval go to zero in the limit results in a rate equation for q_B^A:

$$\dot{q}_B^A = \frac{1}{2} q_B^A \begin{bmatrix} 0 \\ \dot{\underline{\phi}} \end{bmatrix} \tag{3.3.4-7}$$

An expression for the $\dot{\underline{\phi}}$ term in (3.3.4-7) is obtained from the direction cosine matrix equivalent to $q_{B_2}^{B_1}$ which, from Equation (3.2.2.1-8) for small $\underline{\phi}$, is given by:

$$C_{B_2}^{B_1} \approx I + (\underline{\phi} \times) \tag{3.3.4-8}$$

where

I = Identity matrix.

Dividing by the small time interval for movement of B from B_1 to B_2 and letting the time interval go to zero in the limit obtains:

$$\dot{C}_{B_2}^{B_1} \approx (\dot{\underline{\phi}} \times) \tag{3.3.4-9}$$

Applying general Equation (3.3.2-6) (with B_1 substituted for A and B_2 for B), we have:

$$\dot{C}_{B_2}^{B_1} = C_{B_2}^{B_1} \left(\underline{\omega}_{B_1 B_2}^{B_2} \times\right) \tag{3.3.4-10}$$

where

$\underline{\omega}_{B_1 B_2}^{B_2}$ = Angular rate of Frame B_2 relative to Frame B_1 as described in Frame B_2 axes.

But because we are considering Frame B_1 to be stationary relative to Frame A, we can also write:

$$\underline{\omega}_{B_1 B_2}^{B_2} = \underline{\omega}_{A B_2}^{B_2} \tag{3.3.4-11}$$

where

$\underline{\omega}_{A B_2}^{B_2}$ = Angular rate of Frame B_2 relative to Frame A as described in Frame B_2 axes.

For the very small time interval being considered for Frame B_2 motion from Frame B_1, in the limit as the time interval goes to zero, Frames B_1 and B_2 become coincident (i.e., equal to Frame B in general) so that $C_{B_2}^{B_1}$ goes to the identity matrix and $\underline{\omega}_{A B_2}^{B_2}$ goes to $\underline{\omega}_{AB}^{B}$. Hence, with (3.3.4-11) in (3.3.4-10), we obtain:

$$\dot{C}_{B_2}^{B_1} = \left(\underline{\omega}_{AB}^{B} \times\right) \tag{3.3.4-12}$$

Equating (3.3.4-9) and (3.3.4-12) shows that for this particular definition for $\underline{\phi}$:

$$\dot{\underline{\phi}} = \underline{\omega}_{AB}^{B} \tag{3.3.4-13}$$

Substituting Equation (3.3.4-13) into (3.3.4-7) then yields the desired form for the general rate of change of q_B^A as a function of the relative angular rate between Frames A and B:

$$\dot{q}_B^A = \frac{1}{2}\, q_B^A\, \omega_{AB}^{B} \tag{3.3.4-14}$$

in which the following general form applies:

$$\omega_{AB}^{B} \equiv \begin{bmatrix} 0 \\ \underline{\omega}_{AB}^{B} \end{bmatrix} \tag{3.3.4-15}$$

where

ω_{AB}^{B} = Quaternion form of the angular rate vector $\underline{\omega}_{AB}^{B}$.

The rate of change of q_B^A can also be defined in a more familiar form in terms of the individual inertial angular rotation rates of Frames A and B (i.e., angular rotation rates relative to non-rotating inertial space). Using the Equation (3.2.4.1-9) attitude quaternion product chain rule, we first define q_B^A as:

$$q_B^A = q_I^A \, q_B^I \tag{3.3.4-16}$$

where

I = Non-rotating inertial coordinate frame.

The derivative of (3.3.4-16) is:

$$\dot{q}_B^A = q_I^A \, \dot{q}_B^I + \dot{q}_I^A \, q_B^I \tag{3.3.4-17}$$

Applying Equation (3.3.4-14) to the derivative terms in (3.3.4-17) obtains:

$$\dot{q}_B^I = \frac{1}{2} q_B^I \, \omega_{IB}^B \tag{3.3.4-18}$$

and

$$\dot{q}_A^I = \frac{1}{2} q_A^I \, \omega_{IA}^A \tag{3.3.4-19}$$

with

$$\omega_{IB}^B \equiv \begin{bmatrix} 0 \\ \underline{\omega}_{IB}^B \end{bmatrix} \tag{3.3.4-20}$$

$$\omega_{IA}^A \equiv \begin{bmatrix} 0 \\ \underline{\omega}_{IA}^A \end{bmatrix} \tag{3.3.4-21}$$

Based on (3.2.4-8), (3.2.4-14) and the general form of Equation (3.3.4-21), the conjugate of ω_{IA}^A is its negative, hence, with Equations (3.2.4-23) and (3.2.4.1-10), the conjugate of Equation (3.3.4-19) is given by:

$$\dot{q}_I^A = -\frac{1}{2} \omega_{IA}^A \, q_I^A \tag{3.3.4-22}$$

Combining Equations (3.3.4-18) and (3.3.4-22) in (3.3.4-17) yields:

$$\dot{q}_B^A = \frac{1}{2} q_I^A \, q_B^I \, \omega_{IB}^B - \frac{1}{2} \omega_{IA}^A \, q_I^A \, q_B^I \tag{3.3.4-23}$$

Finally, we substitute Equation (3.3.4-16) into (3.3.4-23) to obtain the desired expression for the rate of change of the Frame B to Frame A attitude quaternion as a function of the Frame A and Frame B rotation rates:

$$\dot{q}_B^A = \frac{1}{2}\, q_B^A\, \omega_{IB}^B - \frac{1}{2}\, \omega_{IA}^A\, q_B^A \tag{3.3.4-24}$$

Equation (3.3.4-24) directly parallels the equivalent relationship for the C_B^A direction cosine matrix rate given by Equation (3.3.2-13). Note, however, that Equation (3.3.4-24) is a four-vector equation that must abide by the rules of four-vector multiplication (defined in Section 3.2.4) if it is to be expanded in terms of its components. An important characteristic of Equation (3.3.4-24) is that the coefficient quaternion multiplying the angular rate quaternions is the attitude quaternion itself, hence, from Equation (3.2.4.4-1), its components can never exceed unity in magnitude. Therefore, assuming that the A and B Frames are chosen to have finite angular rates, Equation (3.3.4-24) has no singularities for any attitude of Frame B relative to Frame A.

As an exercise, it is instructive to look at the component form of (3.3.4-24). Substituting the Frame A and B angular rate component definitions from (3.3.2-14) into Equations (3.3.4-20) - (3.3.4-21), and using Equation (3.2.4.1-1) for the Frame B to A attitude quaternion components, Equation (3.3.4-24) in classical matrix form becomes:

$$\begin{bmatrix} \dot{a} \\ \dot{b} \\ \dot{c} \\ \dot{d} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix} \begin{bmatrix} 0 \\ \omega_{XB} \\ \omega_{YB} \\ \omega_{ZB} \end{bmatrix} - \frac{1}{2} \begin{bmatrix} 0 & -\omega_{XA} & -\omega_{YA} & -\omega_{ZA} \\ \omega_{XA} & 0 & -\omega_{ZA} & \omega_{YA} \\ \omega_{YA} & \omega_{ZA} & 0 & -\omega_{XA} \\ \omega_{ZA} & -\omega_{YA} & \omega_{XA} & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \tag{3.3.4-25}$$

or

$$\begin{bmatrix} \dot{a} \\ \dot{b} \\ \dot{c} \\ \dot{d} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -b & -c & -d \\ a & -d & c \\ d & a & -b \\ -c & b & a \end{bmatrix} \begin{bmatrix} \omega_{XB} \\ \omega_{YB} \\ \omega_{ZB} \end{bmatrix} - \frac{1}{2} \begin{bmatrix} -b & -c & -d \\ a & d & -c \\ -d & a & b \\ c & -b & a \end{bmatrix} \begin{bmatrix} \omega_{XA} \\ \omega_{YA} \\ \omega_{ZA} \end{bmatrix} \tag{3.3.4-26}$$

or equivalently:

$$\begin{bmatrix} \dot{a} \\ \dot{b} \\ \dot{c} \\ \dot{d} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -(\omega_{XB} - \omega_{XA}) & -(\omega_{YB} - \omega_{YA}) & -(\omega_{ZB} - \omega_{ZA}) \\ (\omega_{XB} - \omega_{XA}) & 0 & (\omega_{ZB} + \omega_{ZA}) & -(\omega_{YB} + \omega_{YA}) \\ (\omega_{YB} - \omega_{YA}) & -(\omega_{ZB} + \omega_{ZA}) & 0 & (\omega_{XB} + \omega_{XA}) \\ (\omega_{ZB} - \omega_{ZA}) & (\omega_{YB} + \omega_{YA}) & -(\omega_{XB} + \omega_{XA}) & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \tag{3.3.4-27}$$

3.3.5 ROTATION VECTOR RATE EQUATION

Quaternion rate Equation (3.3.4-14) can be used to derive a differential equation for the rate of change of the $\underline{\phi}$ rotation vector describing the relative attitude between arbitrary coordinate Frames A and B, as a function of the relative angular rotation rate between the two frames. For simplicity in the derivation, we will refer to the Equation (3.3.4-14) quaternion q_B^A as "q" and $\underline{\omega}_{AB}^{B}$ in supporting Equation (3.3.4-15) as "$\underline{\omega}$" with magnitude "ω". Furthermore, we will define the quaternion q as a function of its rotation vector equivalent using the Equation (3.2.4.4-2) mixed scalar/vector quaternion notation, i.e.:

$$q = f_1 + f_2\,\underline{\phi} \qquad f_1 \equiv \cos\phi/2 \qquad f_2 \equiv \frac{\sin\phi/2}{\phi} \tag{3.3.5-1}$$

Equation (3.3.4-14) with (3.3.4-15) in Section 3.2.4 mixed scalar/vector notation becomes:

$$\dot{q} = \frac{1}{2}\,q\,\underline{\omega} = \frac{1}{2}\,f_1\,\underline{\omega} + \frac{1}{2}\,f_2\,\underline{\phi}\,\underline{\omega} \tag{3.3.5-2}$$

The $\underline{\phi}\,\underline{\omega}$ vector product in (3.3.5-2) is provided in Section 3.2.4 by Equation (3.2.4-27):

$$\underline{\phi}\,\underline{\omega} = \underline{\phi} \times \underline{\omega} - \underline{\phi} \cdot \underline{\omega} \tag{3.3.5-3}$$

Differentiation of Equations (3.3.5-1) shows that:

$$\begin{aligned}
\dot{q} &= \dot{f}_1 + \dot{f}_2\,\underline{\phi} + f_2\,\dot{\underline{\phi}} \\
\dot{f}_1 &= -\frac{1}{2}\left(\sin\phi/2\right)\dot{\phi} = -\frac{1}{2}\,\phi\,\dot{\phi}\,f_2 \\
\dot{f}_2 &= \frac{1}{2}\,\frac{\cos\phi/2}{\phi}\,\dot{\phi} - \frac{\sin\phi/2}{\phi^2}\,\dot{\phi} = \frac{\dot{\phi}}{\phi}\left(\frac{1}{2}\,f_1 - f_2\right)
\end{aligned} \tag{3.3.5-4}$$

Combining (3.3.5-4) and equating the result to (3.3.5-2) combined with (3.3.5-3) finds:

$$\dot{q} = -\frac{1}{2}\,\phi\,\dot{\phi}\,f_2 + \frac{\dot{\phi}}{\phi}\left(\frac{1}{2}\,f_1 - f_2\right)\underline{\phi} + f_2\,\dot{\underline{\phi}} = \frac{1}{2}\,f_1\,\underline{\omega} + \frac{1}{2}\,f_2\left(\underline{\phi} \times \underline{\omega}\right) - \frac{1}{2}\,f_2\,\underline{\phi} \cdot \underline{\omega} \tag{3.3.5-5}$$

Dividing (3.3.5-5) by f_2 and solving for $\dot{\underline{\phi}}$:

$$\dot{\underline{\phi}} = \frac{1}{2}\frac{f_1}{f_2}\,\underline{\omega} + \frac{1}{2}\left(\underline{\phi} \times \underline{\omega}\right) - \frac{\dot{\phi}}{\phi}\left(\frac{1}{2}\frac{f_1}{f_2} - 1\right)\underline{\phi} + \frac{1}{2}\,\phi\,\dot{\phi} - \frac{1}{2}\,\underline{\phi} \cdot \underline{\omega} \tag{3.3.5-6}$$

Equation (3.3.5-6)) is now separated into its vector and scalar components:

$$\dot{\underline{\phi}} = \frac{1}{2}\frac{f_1}{f_2}\,\underline{\omega} + \frac{1}{2}\left(\underline{\phi} \times \underline{\omega}\right) - \frac{\dot{\phi}}{\phi}\left(\frac{1}{2}\frac{f_1}{f_2} - 1\right)\underline{\phi} \qquad \frac{1}{2}\,\phi\,\dot{\phi} = \frac{1}{2}\,\underline{\phi} \cdot \underline{\omega} \tag{3.3.5-7}$$

The scalar equation is equivalently:

$$\frac{\dot{\phi}}{\phi} = \frac{1}{\phi^2}\,\underline{\phi} \cdot \underline{\omega} \tag{3.3.5-8}$$

Substituting (3.3.5-8) into the vector part of (3.3.5-7) yields:

$$\dot{\underline{\phi}} = \frac{1}{2}\frac{f_1}{f_2}\,\underline{\omega} + \frac{1}{2}\left(\underline{\phi} \times \underline{\omega}\right) - \frac{1}{\phi^2}\left(\frac{1}{2}\frac{f_1}{f_2} - 1\right)\left(\underline{\phi} \cdot \underline{\omega}\right)\underline{\phi} \tag{3.3.5-9}$$

Using the vector triple product rule (Equation (3.1.1-16)), it is easily demonstrated that:

$$\left(\underline{\phi} \cdot \underline{\omega}\right)\underline{\phi} = \underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}\right) + \phi^2\,\underline{\omega} \tag{3.3.5-10}$$

Substituting Equation (3.3.5-10) into (3.3.5-9) obtains:

$$\dot{\underline{\phi}} = \frac{1}{2}\frac{f_1}{f_2}\,\underline{\omega} + \frac{1}{2}\left(\underline{\phi} \times \underline{\omega}\right) + \left(1 - \frac{1}{2}\frac{f_1}{f_2}\right)\underline{\omega} + \frac{1}{\phi^2}\left(1 - \frac{1}{2}\frac{f_1}{f_2}\right)\underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}\right) \tag{3.3.5-11}$$

or, upon combining terms:

$$\dot{\underline{\phi}} = \underline{\omega} + \frac{1}{2}\left(\underline{\phi} \times \underline{\omega}\right) + \frac{1}{\phi^2}\left(1 - \frac{1}{2}\frac{f_1}{f_2}\right)\underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}\right) \tag{3.3.5-12}$$

Using the definitions for f_1 and f_2 in Equations (3.3.5-1)), it can be shown by trigonometric manipulation that the bracketed coefficient in (3.3.5-12) is equivalently:

$$1 - \frac{1}{2}\frac{f_1}{f_2} = 1 - \frac{\phi \sin \phi}{2\,(1 - \cos\,\phi)} \tag{3.3.5-13}$$

Substituting (3.3.5-13) into (3.3.5-12) and reintroducing $\underline{\omega}_{AB}^{B}$ for $\underline{\omega}$ then yields the desired final expression for $\dot{\underline{\phi}}$:

$$\dot{\underline{\phi}} = \underline{\omega}_{AB}^{B} + \frac{1}{2}\,\underline{\phi} \times \underline{\omega}_{AB}^{B} + \frac{1}{\phi^2}\left(1 - \frac{\phi \sin \phi}{2\,(1 - \cos\,\phi)}\right)\underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}_{AB}^{B}\right) \qquad (3.3.5\text{-}14)$$

Equation (3.3.5-14) defines the rate of change of the rotation vector that describes the relative orientation between coordinate Frames A and B as a function of the Frame B relative to Frame A angular rate (as defined in Frame B axes).

3.4 VECTOR RATES OF CHANGE IN ROTATING COORDINATES

Consider an arbitrary vector projected on two arbitrary coordinate frames. Equations (3.2.1-2) show that:

$$\underline{V}^{A} = C_{B}^{A}\,\underline{V}^{B} \qquad (3.4\text{-}1)$$

where

$\underline{V}$ = Arbitrary vector.

A, B = Arbitrary right handed orthogonal coordinate frames.

$\underline{V}^{A}$, $\underline{V}^{B}$ = Arbitrary vector $\underline{V}$ projected on coordinate Frame A, B axes.

C_{B}^{A} = Direction cosine matrix that transforms vectors from Frame B to Frame A.

The rate of change of $\underline{V}$ in Frame A is related to the $\underline{V}$ rate of change in Frame B by the derivative of Equation (3.4-1):

$$\dot{\underline{V}}^{A} = C_{B}^{A}\,\dot{\underline{V}}^{B} + \dot{C}_{B}^{A}\,\underline{V}^{B} \qquad (3.4\text{-}2)$$

Substituting Equation (3.3.2-6) into (3.4-2) then obtains:

$$\dot{\underline{V}}^{A} = C_{B}^{A}\,\dot{\underline{V}}^{B} + C_{B}^{A}\left(\underline{\omega}_{AB}^{B}\times\right)\underline{V}^{B} \qquad (3.4\text{-}3)$$

or

$$\dot{\underline{V}}^{A} = C_{B}^{A}\left(\dot{\underline{V}}^{B} + \underline{\omega}_{AB}^{B} \times \underline{V}^{B}\right) \qquad (3.4\text{-}4)$$

where

$\underline{\omega}_{AB}^{B}$ = Angular rate of Frame B relative to Frame A as projected on Frame B axes.

Equation (3.4-3) can also be expressed in an equivalent form by application of (3.1.1-40) for the $\left(\underline{\omega}_{AB}^{B}\times\right)$ term:

$$\dot{\underline{V}}^A = C_B^A \, \dot{\underline{V}}^B + C_B^A \, C_A^B \left(\underline{\omega}_{AB}^A \times\right) \left(C_A^B\right)^T \underline{V}^B \qquad (3.4\text{-}5)$$

which, with Equations (3.2.1-2) and (3.2.1-3), becomes:

$$\dot{\underline{V}}^A = C_B^A \, \dot{\underline{V}}^B + \underline{\omega}_{AB}^A \times \underline{V}^A \qquad (3.4\text{-}6)$$

where

$\underline{\omega}_{AB}^A$ = Angular rate of Frame B relative to Frame A as projected on Frame A axes.

3.5 ATTITUDE AND VECTOR ERROR CHARACTERISTICS

The error analysis of strapdown inertial navigation systems entails the investigation of error characteristics associated with attitude and vector parameters calculated in the navigation computer. These error effects can arise because of incorrect initialization of data parameters, approximations in digital integration routines for updating the parameters, software programming errors, computer finite word length truncation and round-off effects, and errors in the strapdown inertial sensor data (angular rate sensors and accelerometers) used in calculating the parameters.

The characterization of attitude and vector parameter error effects begins with the definition of an idealized error free attitude or vector parameter and its counterpart implemented in the system computer (hence, containing errors). For this development, the following $\hat{\ }$ notation will be utilized to identify parameters containing errors, where:

$\hat{\ }$ = Indicator of a navigation parameter calculated in the system computer that contains errors. If the idealized (error free) navigation parameter is ( ), then the $\hat{\ }$ version is identified as $(\hat{\ })$.

In general, the difference between $(\hat{\ })$ and ( ) is a measure of the error in $(\hat{\ })$. This section describes generalized methods for applying the previous error definition to characterize error effects associated with attitude and vector parameters.

3.5.1 DIRECTION COSINE MATRIX GENERALIZED ERROR CHARACTERISTICS

Consider a generalized direction cosine matrix in its idealized error free form and in its system computed form where:

C = Idealized error free direction cosine matrix.

$\hat{C}$ = C as calculated in the system computer (i.e., containing errors).

We now define $\hat{C}$ as equal to a variation from C as follows:

$$\hat{C} = (I + E)\,C \tag{3.5.1-1}$$

where

E = Matrix containing $\hat{C}$ errors.

I = Identity matrix.

Since, from Equation (3.2.1-4), the inverse of an idealized direction cosine matrix equals its transpose, Equation (3.5.1-1) is equivalently:

$$E = \hat{C}\,C^T - I \tag{3.5.1-2}$$

where

T = Superscript designation for transpose.

The E matrix can be decomposed into symmetric and skew symmetric elements using the identity:

$$E = \frac{1}{2}\left(E - E^T\right) + \frac{1}{2}\left(E + E^T\right) = F + G \tag{3.5.1-3}$$

with

$$F \equiv \frac{1}{2}\left(E - E^T\right) \qquad G \equiv \frac{1}{2}\left(E + E^T\right) \tag{3.5.1-4}$$

We also note that:

$$F^T = \frac{1}{2}\left(E^T - E\right) = -F \qquad G^T = \frac{1}{2}\left(E^T + E\right) = G \tag{3.5.1-5}$$

Equations (3.5.1-5) show that F is a skew symmetric matrix (i.e., $F_{ij} = -F_{ji}$ and $F_{ii} = 0$) and that G is a symmetric matrix (i.e., $G_{ij} = G_{ji}$), where:

F_{ij}, G_{ij} = Elements in row i and column j of F, G.

Based on the previous results, we rewrite Equations (3.5.1-1) - (3.5.1-4) as follows:

$$\hat{C} = (I + E)\,C \qquad\qquad E = \hat{C}\,C^T - I$$
$$E_{SYM} = \frac{1}{2}\left(E + E^T\right) \qquad\qquad E_{SKSYM} = \frac{1}{2}\left(E - E^T\right) \qquad (3.5.1\text{-}6)$$
$$E = E_{SYM} + E_{SKSYM}$$

where

E_{SYM} = Symmetric portion of E.

E_{SKSYM} = Skew symmetric portion of E.

The interpretation of the E_{SKSYM} portion of E can be ascertained by review of Equation (3.2.2.1-8) for the direction cosine matrix as a function of the rotation vector. If we assume that the $\underline{\phi}$ rotation vector is small, Equation (3.2.2.1-8) can be approximated to first order by:

$$C_{A_2}^{A_1} \approx I + \left(\underline{\phi}\times\right) \qquad (3.5.1\text{-}7)$$

where

A_1, A_2 = Arbitrary coordinate frames that are angularly close to one another.

Thus, to first order, a small angle direction cosine matrix has the general form of identity plus a skew symmetric matrix formed from its small angle rotation vector. The (I + E) term in the Equations (3.5.1-6) $\hat{C}$ expression has exactly the same form as Equation (3.5.1-7) when E is replaced by E_{SKSYM}. We can conclude then that the E_{SKSYM} component of E in Equations (3.5.1-6) has the equivalent effect of adding a rotation error vector to C with components defined by the off-diagonal elements of E_{SKSYM}. In other words, if C represents the angular attitude between two general coordinate Frames A and B, then the effect of E_{SKSYM} on $\hat{C}$ is to rotate Frame B relative to Frame A by the small angle rotation vector formed from the components of E_{SKSYM}. In effect, then, E_{SKSYM} represents misalignment in the $\hat{C}$ matrix.

In order to investigate the characteristics of the E_{SYM} portion of E, we analyze the product $\hat{C}\,\hat{C}^T$ using $\hat{C}$ from Equations (3.5.1-6) with E from the last expression in (3.5.1-6):

$$\begin{aligned}
\hat{C}\,\hat{C}^T &= \left(I + E_{SYM} + E_{SKSYM}\right) C \left[\left(I + E_{SYM} + E_{SKSYM}\right) C\right]^T \\
&= \left(I + E_{SYM} + E_{SKSYM}\right) C\, C^T \left(I + E_{SYM} + E_{SKSYM}\right)^T \\
&= \left(I + E_{SYM} + E_{SKSYM}\right) C\, C^{-1} \left(I + E_{SYM} - E_{SKSYM}\right) \qquad (3.5.1\text{-}8) \\
&= \left(I + E_{SYM} + E_{SKSYM}\right)\left(I + E_{SYM} - E_{SKSYM}\right) \\
&= I + 2\,E_{SYM} + \text{second order terms}
\end{aligned}$$

The Equation (3.5.1-8) development made use of generalized Equations (3.2.1-4) which states that the transpose of an idealized direction cosine matrix equals its inverse. Dropping the second order terms in (3.5.1-8) as negligible, then yields for E_{SYM}:

$$E_{SYM} = \frac{1}{2}\left(\hat{C}\,\hat{C}^T - I\right) \tag{3.5.1-9}$$

Equation (3.5.1-9) can be interpreted by expansion of $\hat{C}$:

$$\hat{C}^T = \begin{bmatrix} \hat{\underline{u}}_1 & \hat{\underline{u}}_2 & \hat{\underline{u}}_3 \end{bmatrix} \qquad \hat{C} = \begin{bmatrix} \hat{\underline{u}}_1^T \\ \hat{\underline{u}}_2^T \\ \hat{\underline{u}}_3^T \end{bmatrix} \tag{3.5.1-10}$$

where

$\hat{\underline{u}}_1, \hat{\underline{u}}_2, \hat{\underline{u}}_3$ = Columns of $\hat{C}^T$ (i.e., the rows of $\hat{C}$).

Substituting (3.5.1-10) into (3.5.1-9) and applying (3.1.1-12) yields:

$$\begin{aligned} E_{SYM} &= \frac{1}{2}\begin{bmatrix} \hat{\underline{u}}_1^T\,\hat{\underline{u}}_1 - 1 & \hat{\underline{u}}_1^T\,\hat{\underline{u}}_2 & \hat{\underline{u}}_1^T\,\hat{\underline{u}}_3 \\ \hat{\underline{u}}_2^T\,\hat{\underline{u}}_1 & \hat{\underline{u}}_2^T\,\hat{\underline{u}}_2 - 1 & \hat{\underline{u}}_2^T\,\hat{\underline{u}}_3 \\ \hat{\underline{u}}_3^T\,\hat{\underline{u}}_1 & \hat{\underline{u}}_3^T\,\hat{\underline{u}}_2 & \hat{\underline{u}}_3^T\,\hat{\underline{u}}_3 - 1 \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} \left(\hat{\underline{u}}_1 \cdot \hat{\underline{u}}_1 - 1\right) & \left(\hat{\underline{u}}_1 \cdot \hat{\underline{u}}_2\right) & \left(\hat{\underline{u}}_1 \cdot \hat{\underline{u}}_3\right) \\ \left(\hat{\underline{u}}_2 \cdot \hat{\underline{u}}_1\right) & \left(\hat{\underline{u}}_2 \cdot \hat{\underline{u}}_2 - 1\right) & \left(\hat{\underline{u}}_2 \cdot \hat{\underline{u}}_3\right) \\ \left(\hat{\underline{u}}_3 \cdot \hat{\underline{u}}_1\right) & \left(\hat{\underline{u}}_3 \cdot \hat{\underline{u}}_2\right) & \left(\hat{\underline{u}}_3 \cdot \hat{\underline{u}}_3 - 1\right) \end{bmatrix} \end{aligned} \tag{3.5.1-11}$$

Equations (3.2.1-6) show that for the idealized direction cosine matrix C, the rows represent unit vectors along orthogonal reference coordinate frame axes. Hence, the rows of C (treating the rows as vectors) are unity in magnitude ("normal") and orthogonal to one another. We will now show from (3.5.1-11), that E_{SYM} measures the degree of orthogonality and normality error in $\hat{C}$.

The orthogonality error between any two rows of $\hat{C}$ (with the rows treated as vectors) can be measured by the dot product between the rows which should be zero if the rows are perpendicular to one another (see Equation (3.1.1-2)):

$$\hat{\underline{u}}_i \cdot \hat{\underline{u}}_j = \cos\left(\frac{\pi}{2} + \delta_{Orth_{ij}}\right) = \sin \delta_{Orth_{ij}} \approx \delta_{Orth_{ij}} \tag{3.5.1-12}$$

where

$\hat{\underline{u}}_i$ = Column i of $\hat{C}^T$ (or row i of $\hat{C}$).

$\delta_{Orth\ ij}$ = Orthogonality error between rows i and j of $\hat{C}$.

The normality error for any row of $\hat{C}$ can be measured by the dot product between the row and itself (the magnitude squared - See Equation (3.1.1-2)) compared with one, and can be evaluated based on the definition:

$$\hat{\underline{u}}_i = \left(1 + \delta_{Norm\ i}\right) \underline{u}_i \tag{3.5.1-13}$$

where

$\underline{u}_i$ = Column i of C^T (or row i of C) which is unity in magnitude and orthogonal to any other row of C.

$\delta_{Norm\ i}$ = Normality error in $\hat{\underline{u}}_i$.

Taking the dot product between (3.5.1-13) and itself and subtracting one shows that:

$$\hat{\underline{u}}_i \cdot \hat{\underline{u}}_i - 1 = \left(1 + \delta_{Norm_i}\right)^2 \underline{u}_i \cdot \underline{u}_i - 1 = \left(1 + \delta_{Norm_i}\right)^2 - 1 \approx 2\, \delta_{Norm_i} \tag{3.5.1-14}$$

Comparing (3.5.1-12) and (3.5.1-14) with (3.5.1-11) we see that:

$$\varepsilon_{ij} = \frac{1}{2}\left(\hat{\underline{u}}_i \cdot \hat{\underline{u}}_j\right) = \frac{1}{2}\, \delta_{Orth_{ij}} \qquad \varepsilon_{ii} = \frac{1}{2}\left(\hat{\underline{u}}_i \cdot \hat{\underline{u}}_i - 1\right) = \delta_{Norm_i} \tag{3.5.1-15}$$

where

ε_{ij} = Element in row i, column j of E_{SYM}.

Thus, from (3.5.1-15), the off-diagonal element in row i, column j of E_{SYM} equals half the orthogonality error between rows i and j of $\hat{C}$, while the i^{th} diagonal element of E_{SYM} equals the normality error in row i of $\hat{C}$.

It is also instructive to analyze the orthogonality/normality characteristics of the $\hat{C}$ columns. The procedure is identical to that leading to Equation (3.5.1-15) except that Equations (3.5.1-1) and (3.5.1-2) would be replaced by:

$$\hat{C} = C\,(I + E') \qquad E' = C^T\,\hat{C} - I \tag{3.5.1-16}$$

where

E' = Alternative definition for the matrix containing $\hat{C}$ errors.

Then, following the same steps leading to (3.5.1-9), we would find that:

$$E'_{SYM} = \frac{1}{2}\left(\hat{C}^T\,\hat{C} - I\right) \tag{3.5.1-17}$$

where

E'_{SYM} = Symmetric portion of E'.

Comparing (3.5.1-17) for E'_{SYM} with (3.5.1-9) for E_{SYM}, we see that they are of identical form but with $\hat{C}$ replaced by $\hat{C}^T$. We can conclude, therefore, that since E_{SYM} represents the orthogonality/normality error in the rows of $\hat{C}$ (as shown in (3.5.1-15)), that E'_{SYM} represents the orthogonality/normality error in the columns of $\hat{C}$.

We also note from Equations (3.5.1-9) and (3.5.1-17) that:

$$E_{SYM}\,\hat{C} = \frac{1}{2}\left(\hat{C}\,\hat{C}^T - I\right)\hat{C} = \hat{C}\,\frac{1}{2}\left(\hat{C}^T\,\hat{C} - I\right) = \hat{C}\,E'_{SYM} \tag{3.5.1-18}$$

Equation (3.5.1-18) shows that if the rows of $\hat{C}$ are normal and orthogonal (i.e., $E_{SYM} = 0$), then the columns of $\hat{C}$ will also be normal and orthogonal (i.e., $E'_{SYM} = 0$).

For the remainder of this section, we shall analyze the dynamic properties of E_{SYM}, E'_{SYM} and E_{SKSYM} under angular motion of the coordinate frames defining C and $\hat{C}$. The analysis is expedited if we become more specific regarding our definition for C to be the C_B^A matrix of Section 3.3.2. Then Equation (3.3.2-13) applies for which:

$$\dot{C}_B^A = C_B^A\left(\underline{\omega}_{IB}^B\times\right) - \left(\underline{\omega}_{IA}^A\times\right)C_B^A \tag{3.5.1-19}$$

where

$\underline{\omega}_{IB}^B$ = Angular rate of Frame B relative to a non-rotating inertial Frame I (as projected on Frame B axes).

$\underline{\omega}_{IA}^A$ = Angular rate of Frame A relative to inertial Frame I (as projected on Frame A axes).

The version of (3.5.1-19) implemented in the strapdown system computer can be represented as:

$$\dot{\hat{C}}_B^A = \hat{C}_B^A \left(\hat{\underline{\omega}}_{IB}^B \times\right) - \left(\hat{\underline{\omega}}_{IA}^A \times\right) \hat{C}_B^A + \delta\dot{C}_{B_{Comp}}^A \tag{3.5.1-20}$$

where

$\delta\dot{C}_{B_{Comp}}^A$ = Error rate input to $\dot{\hat{C}}_B^A$ created by approximations in the digital integration routines used for calculating $\hat{C}$, software programming errors, and computer finite word length truncation/round-off effects.

$\hat{\underline{\omega}}_{IA}^A$, $\hat{\underline{\omega}}_{IB}^B$ = Values for $\underline{\omega}_{IA}^A$, $\underline{\omega}_{IB}^B$ in the computer executing (3.5.1-20) that may contain errors compared to the error free $\underline{\omega}_{IA}^A$, $\underline{\omega}_{IB}^B$ values.

Let us now analyze the rate of change of E_{SYM} (i.e., the orthogonality/normality error in the rows of $\hat{C}_B^A$) under motion characterized by Equations (3.5.1-19) - (3.5.1-20). From Equation (3.5.1-9) we first write:

$$E_{SYM} = \frac{1}{2}\left[\hat{C}_B^A \left(\hat{C}_B^A\right)^T - I\right] \tag{3.5.1-21}$$

whose derivative is:

$$\dot{E}_{SYM} = \frac{1}{2}\left[\dot{\hat{C}}_B^A \left(\hat{C}_B^A\right)^T + \hat{C}_B^A \left(\dot{\hat{C}}_B^A\right)^T\right] \tag{3.5.1-22}$$

Using the property that the transpose of a cross-product operator equals its negative, the transpose of (3.5.1-20) for Equation (3.5.1-22) is:

$$\left(\dot{\hat{C}}_B^A\right)^T = - \left(\hat{\underline{\omega}}_{IB}^B \times\right)\left(\hat{C}_B^A\right)^T + \left(\hat{C}_B^A\right)^T \left(\hat{\underline{\omega}}_{IA}^A \times\right) + \left(\delta\dot{C}_{B_{Comp}}^A\right)^T \tag{3.5.1-23}$$

Substituting (3.5.1-20) and (3.5.1-23) into (3.5.1-22) then gives:

$$\begin{aligned}
2\,\dot{E}_{SYM} &= \left[\hat{C}_B^A \left(\hat{\underline{\omega}}_{IB}^B \times\right) - \left(\hat{\underline{\omega}}_{IA}^A \times\right) \hat{C}_B^A + \delta\dot{C}_{B_{Comp}}^A\right]\left(\hat{C}_B^A\right)^T \\
&\qquad + \hat{C}_B^A \left[- \left(\hat{\underline{\omega}}_{IB}^B \times\right)\left(\hat{C}_B^A\right)^T + \left(\hat{C}_B^A\right)^T \left(\hat{\underline{\omega}}_{IA}^A \times\right) + \left(\delta\dot{C}_{B_{Comp}}^A\right)^T\right] \\
&= \hat{C}_B^A \left(\hat{\underline{\omega}}_{IB}^B \times\right)\left(\hat{C}_B^A\right)^T - \left(\hat{\underline{\omega}}_{IA}^A \times\right) \hat{C}_B^A \left(\hat{C}_B^A\right)^T + \delta\dot{C}_{B_{Comp}}^A \left(\hat{C}_B^A\right)^T \qquad (3.5.1\text{-}24) \\
&\qquad - \hat{C}_B^A \left(\hat{\underline{\omega}}_{IB}^B \times\right)\left(\hat{C}_B^A\right)^T + \hat{C}_B^A \left(\hat{C}_B^A\right)^T \left(\hat{\underline{\omega}}_{IA}^A \times\right) + \hat{C}_B^A \left(\delta\dot{C}_{B_{Comp}}^A\right)^T \\
&= \hat{C}_B^A \left(\hat{C}_B^A\right)^T \left(\hat{\underline{\omega}}_{IA}^A \times\right) - \left(\hat{\underline{\omega}}_{IA}^A \times\right) \hat{C}_B^A \left(\hat{C}_B^A\right)^T + \delta\dot{C}_{B_{Comp}}^A \left(\hat{C}_B^A\right)^T + \hat{C}_B^A \left(\delta\dot{C}_{B_{Comp}}^A\right)^T
\end{aligned}$$

or with (3.5.1-21) for the $\hat{C}_B^A \left(\hat{C}_B^A\right)^T$ terms:

$$\dot{E}_{SYM} = E_{SYM}\left(\hat{\underline{\omega}}_{IA}^A \times\right) - \left(\hat{\underline{\omega}}_{IA}^A \times\right) E_{SYM} + \frac{1}{2}\left[\delta\dot{C}_{B_{Comp}}^A \left(\hat{C}_B^A\right)^T + \hat{C}_B^A \left(\delta\dot{C}_{B_{Comp}}^A\right)^T\right] \qquad (3.5.1\text{-}25)$$

Equation (3.5.1-25) clearly shows that in the absence of computational error (i.e., for $\delta\dot{C}_{B_{Comp}}^A = 0$), E_{SYM} will be zero if it is initialized at zero (i.e., if $\hat{C}_B^A$ is initialized with zero orthogonality/normalization error). Thus, we can conclude that E_{SYM} can only be created by initial $\hat{C}_B^A$ orthogonality/normality error or by $\delta\dot{C}_{B_{Comp}}^A$ inputs (due to approximations in the digital integration routines used for calculating $\hat{C}_B^A$, software programming errors, and computer finite word length truncation/round-off effects), but <u>not</u> from errors in the $\hat{\underline{\omega}}_{IA}^A$, $\hat{\underline{\omega}}_{IB}^B$ angular rates applied in the $\hat{C}_B^A$ integration operation.

The rate of change of E'_{SYM} (the orthogonality/normality error in the columns of $\hat{C}_B^A$) is obtained similarly beginning with Equation (3.5.1-17). The result is:

$$\dot{E}'_{SYM} = E'_{SYM}\left(\hat{\underline{\omega}}_{IB}^B \times\right) - \left(\hat{\underline{\omega}}_{IB}^B \times\right) E'_{SYM} + \frac{1}{2}\left[\left(\hat{C}_B^A\right)^T \delta\dot{C}_{B_{Comp}}^A + \left(\delta\dot{C}_{B_{Comp}}^A\right)^T \hat{C}_B^A\right] \qquad (3.5.1\text{-}26)$$

Conclusions from (3.5.1-26) regarding the cause of E'_{SYM} are the same as the discussion following Equation (3.5.1-25) for the E_{SYM} error.

Finally, we analyze E_{SKSYM}, the misalignment error associated with $\hat{C}_B^A$, under dynamic motion. Substituting the E expression in Equations (3.5.1-6) into the E_{SKSYM} expression, and introducing the more specific C_B^A for C we see that:

$$E_{SKSYM} = \frac{1}{2}\left[\hat{C}_B^A \left(C_B^A\right)^T - C_B^A \left(\hat{C}_B^A\right)^T\right] \qquad (3.5.1\text{-}27)$$

whose derivative is:

$$\dot{E}_{SKSYM} = \frac{1}{2}\left[\dot{\hat{C}}_B^A \left(C_B^A\right)^T + \hat{C}_B^A \left(\dot{C}_B^A\right)^T - \dot{C}_B^A \left(\hat{C}_B^A\right)^T - C_B^A \left(\dot{\hat{C}}_B^A\right)^T\right] \qquad (3.5.1\text{-}28)$$

The $\left(\dot{C}_B^A\right)^T$ term in (3.5.1-28) is the transpose of (3.5.1-19):

$$\left(\dot{C}_B^A\right)^T = -\left(\underline{\omega}_{IB}^B \times\right)\left(C_B^A\right)^T + \left(C_B^A\right)^T \left(\underline{\omega}_{IA}^A \times\right) \qquad (3.5.1\text{-}29)$$

Substituting (3.5.1-19), (3.5.1-20), (3.5.1-23) and (3.5.1-29) into (3.5.1-28) (multiplied by 2) then yields:

$$
\begin{aligned}
2\,\dot{E}_{SKSYM} = &\left[\hat{C}_B^A\left(\underline{\hat{\omega}}_{IB}^B\times\right) - \left(\underline{\hat{\omega}}_{IA}^A\times\right)\hat{C}_B^A + \delta\dot{C}_{B_{Comp}}^A\right]\left(C_B^A\right)^T \\
&+ \hat{C}_B^A\left[-\left(\underline{\omega}_{IB}^B\times\right)\left(C_B^A\right)^T + \left(C_B^A\right)^T\left(\underline{\omega}_{IA}^A\times\right)\right] - \left[C_B^A\left(\underline{\omega}_{IB}^B\times\right) - \left(\underline{\omega}_{IA}^A\times\right)C_B^A\right]\left(\hat{C}_B^A\right)^T \\
&- C_B^A\left[-\left(\underline{\hat{\omega}}_{IB}^B\times\right)\left(\hat{C}_B^A\right)^T + \left(\hat{C}_B^A\right)^T\left(\underline{\hat{\omega}}_{IA}^A\times\right) + \left(\delta\dot{C}_{B_{Comp}}^A\right)^T\right] \\
= &\left\{\hat{C}_B^A\left[\left(\underline{\omega}_{IB}^B + \delta\underline{\omega}_{IB}^B\right)\times\right] - \left(\underline{\hat{\omega}}_{IA}^A\times\right)\hat{C}_B^A + \delta\dot{C}_{B_{Comp}}^A\right\}\left(C_B^A\right)^T \\
&+ \hat{C}_B^A\left\{-\left(\underline{\omega}_{IB}^B\times\right)\left(C_B^A\right)^T + \left(C_B^A\right)^T\left[\left(\underline{\hat{\omega}}_{IA}^A - \delta\underline{\omega}_{IA}^A\right)\times\right]\right\} \\
&- \left\{C_B^A\left(\underline{\omega}_{IB}^B\times\right) - \left[\left(\underline{\hat{\omega}}_{IA}^A - \delta\underline{\omega}_{IA}^A\right)\times\right]C_B^A\right\}\left(\hat{C}_B^A\right)^T \\
&- C_B^A\left\{-\left[\left(\underline{\omega}_{IB}^B + \delta\underline{\omega}_{IB}^B\right)\times\right]\left(\hat{C}_B^A\right)^T + \left(\hat{C}_B^A\right)^T\left(\underline{\hat{\omega}}_{IA}^A\times\right) + \left(\delta\dot{C}_{B_{Comp}}^A\right)^T\right\} \\
= &\;\hat{C}_B^A\left(\delta\underline{\omega}_{IB}^B\times\right)\left(C_B^A\right)^T + C_B^A\left(\delta\underline{\omega}_{IB}^B\times\right)\left(\hat{C}_B^A\right)^T - \hat{C}_B^A\left(C_B^A\right)^T\left(\delta\underline{\omega}_{IA}^A\times\right) - \left(\delta\underline{\omega}_{IA}^A\times\right)C_B^A\left(\hat{C}_B^A\right)^T \\
&+ \left[\hat{C}_B^A\left(C_B^A\right)^T - C_B^A\left(\hat{C}_B^A\right)^T\right]\left(\underline{\hat{\omega}}_{IA}^A\times\right) - \left(\underline{\hat{\omega}}_{IA}^A\times\right)\left[\hat{C}_B^A\left(C_B^A\right)^T - C_B^A\left(\hat{C}_B^A\right)^T\right] \\
&+ \delta\dot{C}_{B_{Comp}}^A\left(C_B^A\right)^T - C_B^A\left(\delta\dot{C}_{B_{Comp}}^A\right)^T
\end{aligned}
\tag{3.5.1-30}
$$

in which

$$
\delta\underline{\omega}_{IB}^B \equiv \underline{\hat{\omega}}_{IB}^B - \underline{\omega}_{IB}^B \qquad\qquad \delta\underline{\omega}_{IA}^A \equiv \underline{\hat{\omega}}_{IA}^A - \underline{\omega}_{IA}^A
\tag{3.5.1-31}
$$

where

$$
\delta\underline{\omega}_{IB}^B,\ \delta\underline{\omega}_{IA}^A = \text{Errors in } \underline{\hat{\omega}}_{IB}^B,\ \underline{\hat{\omega}}_{IA}^A .
$$

Equation (3.5.1-30) can be further reduced by approximating $\hat{C}_B^A$ as C_B^A in the $\delta\underline{\omega}_{IB}^B$, $\delta\underline{\omega}_{IA}^A$ terms, substituting (3.5.1-27) in the $\underline{\hat{\omega}}_{IA}^A$ terms, and applying similarity transformation Equation (3.1.1-38). The final result (divided by 2) then is:

$$
\begin{aligned}
\dot{E}_{SKSYM} = &\left[\left(C_B^A\,\delta\underline{\omega}_{IB}^B\right)\times\right] - \left(\delta\underline{\omega}_{IA}^A\times\right) + E_{SKSYM}\left(\underline{\hat{\omega}}_{IA}^A\times\right) - \left(\underline{\hat{\omega}}_{IA}^A\times\right)E_{SKSYM} \\
&+ \frac{1}{2}\left[\delta\dot{C}_{B_{Comp}}^A\left(C_B^A\right)^T - C_B^A\left(\delta\dot{C}_{B_{Comp}}^A\right)^T\right]
\end{aligned}
\tag{3.5.1-32}
$$

Equation (3.5.1-32) shows that E_{SKSYM} (the misalignment error in $\hat{C}_B^A$) is created by initial $\hat{C}_B^A$ misalignment error, by $\delta\dot{C}_{B_{Comp}}^A$ inputs (due to approximations in digital integration routines for calculating $\hat{C}_B^A$, software programming errors, and computer finite word length truncation/round-off effects), and unlike E_{SYM} and E'_{SYM}, by errors in the $\hat{\underline{\omega}}_{IA}^A$, $\hat{\underline{\omega}}_{IB}^B$ angular rates used in the $\hat{C}_B^A$ integration routine.

3.5.2 DIRECTION COSINE MATRIX MISALIGNMENT ERROR CHARACTERISTICS

Consider a general direction cosine matrix that transforms vectors from a general coordinate Frame B Frame to general coordinate Frame A. Consider that this direction cosine matrix is being calculated in the strapdown inertial navigation system computer, hence may contain errors. We define the error in the direction cosine matrix as:

$$\delta C_B^A \equiv \hat{C}_B^A - C_B^A \tag{3.5.2-1}$$

where

C_B^A = Direction cosine matrix that transform vectors from the B Frame to the A Frame.

$\hat{C}_B^A$ = C_B^A as calculated in the system computer.

$\hat{}$ = Designator for a system computer calculated quantity, hence, containing error. The quantity without the $\hat{}$ designation is by definition error free, hence, C_B^A is error free and $\hat{C}_B^A$ contains errors.

δC_B^A = Error in $\hat{C}_B^A$.

From Section 3.5.1, the errors contained in $\hat{C}_B^A$ can be classified as normalization, orthogonality and misalignment errors. The normalization and orthogonality errors are created by initialization error, software computation algorithm error and/or finite word length computer error. As discussed in Section 3.5.1, normalization and orthogonality errors are not created by strapdown inertial sensor errors present on the signals used to compute the $\hat{C}_B^A$ matrix. In general, proper software design practice will assure that the normalization/orthogonality errors in $\hat{C}_B^A$ will be negligibly small. Therefore, the $\hat{C}_B^A$ error during normal inertial navigation system operation will be primarily misalignment error. For properly designed software (i.e., with negligible algorithm error, negligible finite computer word-length error, and no programming error), the remaining $\hat{C}_B^A$ misalignment error will be produced by $\hat{C}_B^A$ initialization error and by errors in the strapdown inertial sensor inputs used in the $\hat{C}_B^A$ calculation/updating process.

For this section, we will assume that the design of the basic strapdown system has been performed properly so that the normalization/orthogonality error in software computed direction cosine matrices is negligibly small. Hence, we will only be dealing with the direction cosine matrix misalignment errors. As such, the general direction cosine matrices $\hat{C}_B^A$ and C_B^A will both be considered to be orthogonal and normal, with the δC_B^A error in $\hat{C}_B^A$ only containing the effects of misalignment. Section 3.5.1 showed that the normalization and orthogonality errors in a general strapdown software computed direction cosine matrix $\hat{C}$ are contained in the symmetrical portion of $\hat{C}$. If we assume zero normality/orthogonality error, the symmetric error portion of $\hat{C}$ (i.e., E_{SYM}) will be zero, hence with $\hat{C}_B^A$ for $\hat{C}$ in Equation (3.5.1-9):

$$E_{SYM} = \frac{1}{2}\left[\hat{C}_B^A \left(\hat{C}_B^A\right)^T - I\right] = 0 \tag{3.5.2-2}$$

or upon rearrangement:

$$\hat{C}_B^A \left(\hat{C}_B^A\right)^T = I \tag{3.5.2-3}$$

Equations (3.5.1-6) define the strapdown software computed $\hat{C}_B^A$ matrix as a variation from the correct C_B^A value:

$$\hat{C}_B^A = (I + E)\, C_B^A \tag{3.5.2-4}$$

Substituting (3.5.2-4) into Equation (3.5.2-3) using the (3.2.1-4) property of the idealized C_B^A that its inverse equals its transpose, then obtains:

$$\hat{C}_B^A \left(\hat{C}_B^A\right)^T = I = (I + E)\; C_B^A \left(C_B^A\right)^T (I + E)^T = (I + E)\,(I + E)^T \tag{3.5.2-5}$$

or

$$(I + E)^T = (I + E)^{-1} \tag{3.5.2-6}$$

Equation (3.5.2-6) shows that the transpose of (I + E) in Equation (3.5.2-4) equals its inverse. This is exactly the characteristic of an idealized direction cosine matrix (see Equation (3.2.1-4)). We can conclude that under the zero orthogonality/normality condition, (I + E) in (3.5.2-4) can be treated as a direction cosine matrix containing the misalignment errors in $\hat{C}_B^A$. This is the approach we will take for the remainder of this chapter in treating direction cosine matrix error characteristics.

Since $\hat{C}_B^A$ defines the <u>relative</u> attitude between Frames A and B, we can arbitrarily assign the misalignment error in $\hat{C}_B^A$ to an error in B Frame attitude relative to Frame A, or to an error in A Frame attitude relative to Frame B. Assuming the latter for the moment, we then write:

$$\hat{C}_B^A = C_B^{\hat{A}} \tag{3.5.2-7}$$

where

$\hat{A}$ = A Frame attitude assumed to be in error from the correct A Frame attitude.

Using the Equation (3.2.1-5) chain rule, we can also write:

$$\hat{C}_B^A = C_A^{\hat{A}}\, C_B^A \tag{3.5.2-8}$$

where

$C_A^{\hat{A}}$ = Direction cosine matrix that would transform vectors from the error free A Frame attitude to the misaligned $\hat{A}$ Frame attitude.

Note that Equation (3.5.2-8) is equivalent to the Equation (3.5.2-4) form in which (I + E) has been replaced by the $C_A^{\hat{A}}$ direction cosine matrix. Substituting (3.5.2-8) into (3.5.2-1) then yields for the $\hat{C}_B^A$ error:

$$\delta C_B^A = C_A^{\hat{A}}\, C_B^A - C_B^A = \left(C_A^{\hat{A}} - I\right) C_B^A \tag{3.5.2-9}$$

The $C_A^{\hat{A}}$ matrix can also be represented by a rotation error vector. Convention defines this rotation error vector from the inverse of $C_A^{\hat{A}}$ (i.e., from $C_{\hat{A}}^A$). Assuming that the $\hat{A}$ and A Frames are almost parallel (i.e., only a small angle attitude error in the $\hat{A}$ Frame attitude), generalized Equation (3.2.2.1-8) shows that to first order, $C_{\hat{A}}^A$ can be approximated as:

$$C_{\hat{A}}^A \approx I + \left(\underline{\alpha}\times\right) \tag{3.5.2-10}$$

where

$\underline{\alpha}$ = Rotation error vector associated with $C_{\hat{A}}^{A}$.

Taking the transpose of (3.5.2-10), recognizing from (3.1.1-14) that the transpose of a cross-product operator equals its negative, and applying (3.1-12), then yields for $C_{A}^{\hat{A}}$:

$$C_{A}^{\hat{A}} = I - \left(\underline{\alpha}\times\right) \tag{3.5.2-11}$$

Substitution of (3.5.2-11) into (3.5.2-8) (recognizing from Equations (3.2.1-4) that the inverse of an idealized direction cosine matrix equals its transpose) shows after rearrangement, that Equation (3.5.2-11) is equivalent to:

$$\left(\underline{\alpha}\times\right) = I - \hat{C}_{B}^{A}\left(C_{B}^{A}\right)^{T} \tag{3.5.2-12}$$

Substituting (3.5.2-11) into (3.5.2-9) provides an expression for the $\hat{C}_{B}^{A}$ error as a function of the $C_{\hat{A}}^{A}$ rotation error vector:

$$\delta C_{B}^{A} = -\left(\underline{\alpha}\times\right) C_{B}^{A} \tag{3.5.2-13}$$

The previous analysis was based on the assumption that A Frame attitude error was the source of the error in $\hat{C}_{B}^{A}$. The analysis can also be performed based on the assumption that B Frame attitude error is the source of $\hat{C}_{B}^{A}$ error, i.e.:

$$\hat{C}_{B}^{A} = C_{\hat{B}}^{A} \tag{3.5.2-14}$$

where

$\hat{B}$ = B Frame attitude assumed to be in error from the correct B Frame attitude.

Using the Equation (3.2.1-5) chain rule, we write:

$$\hat{C}_{B}^{A} = C_{B}^{A}\, C_{\hat{B}}^{B} \tag{3.5.2-15}$$

where

$C_{\hat{B}}^{B}$ = Direction cosine matrix that would transform vectors from the $\hat{B}$ Frame error attitude to the error free B Frame attitude.

Substituting (3.5.2-15) into (3.5.2-1) then yields for the $\hat{C}_{B}^{A}$ error:

$$\delta C_B^A \;=\; C_B^A\, C_{\hat{B}}^B - C_B^A \;=\; C_B^A\left(C_{\hat{B}}^B - I\right) \tag{3.5.2-16}$$

Defining $C_{\hat{B}}^B$ in terms of a small rotation error vector obtains:

$$C_{\hat{B}}^B \;\equiv\; I + \left(\underline{\beta}\times\right) \tag{3.5.2-17}$$

where

$\underline{\beta}$ = Rotation error vector associated with $C_{\hat{B}}^B$.

Substituting (3.5.2-17) into (3.5.2-15) then yields after rearrangement:

$$\left(\underline{\beta}\times\right) \;=\; \left(C_B^A\right)^T \hat{C}_B^A - I \tag{3.5.2-18}$$

Substitution of (3.5.2-17) in (3.5.2-16) provides an expression for the $\hat{C}_B^A$ error as a function of the $C_{\hat{B}}^B$ rotation error vector:

$$\delta C_B^A \;=\; C_B^A\left(\underline{\beta}\times\right) \tag{3.5.2-19}$$

Equation (3.5.2-13) with (3.5.2-12) and Equation (3.5.2-19) with (3.5.2-18) are two equally valid methods for defining the error in the $\hat{C}_B^A$ matrix in terms of the $\underline{\alpha}$ or $\underline{\beta}$ rotation error vectors. In order to find the relationship between $\underline{\alpha}$ and $\underline{\beta}$ we equate Equations (3.5.2-13) and (3.5.2-19):

$$\left(\underline{\alpha}\times\right) C_B^A \;=\; -\, C_B^A\left(\underline{\beta}\times\right) \tag{3.5.2-20}$$

or, after rearrangement with (3.2.1-3) and (3.2.1-4):

$$\left(\underline{\alpha}\times\right) \;=\; -\, C_B^A\left(\underline{\beta}\times\right)\left(C_B^A\right)^T \qquad\qquad \left(\underline{\beta}\times\right) \;=\; -\, C_A^B\left(\underline{\alpha}\times\right)\left(C_A^B\right)^T \tag{3.5.2-21}$$

Applying general Equation (3.2.1-8) with (3.2.1-2) to the $\left(\underline{\alpha}\times\right)$ expression in (3.5.2-21) then yields:

$$\left(\underline{\alpha}\times\right) \;=\; -\left[\left(C_B^A\,\underline{\beta}\right)\times\right] \tag{3.5.2-22}$$

or, finally:

$$\underline{\alpha} \;=\; -\, C_B^A\,\underline{\beta} \tag{3.5.2-23}$$

The same procedure applied to $(\underline{\beta}\times)$ in (3.5.2-21) shows that:

$$\underline{\beta} = -\left(C_B^A\right)^T \underline{\alpha} \tag{3.5.2-24}$$

Equations (3.5.2-23) and (3.5.2-24) are alternative forms of Equations (3.5.2-21) which clearly show that $\underline{\alpha}$ is the negative of $\underline{\beta}$ in Frame A and $\underline{\beta}$ is the negative of $\underline{\alpha}$ viewed in Frame B. For clarity, then, we define:

$$\underline{\alpha}_{B\,to\,A}^{A} \equiv \underline{\alpha} \qquad \underline{\beta}_{B\,to\,A}^{B} \equiv \underline{\beta} \tag{3.5.2-25}$$

and from (3.5.2-23) and (3.5.2-24) with (3.2.1-2):

$$\underline{\alpha}_{B\,to\,A}^{B} = -\,\underline{\beta}_{B\,to\,A}^{B} \qquad \underline{\beta}_{B\,to\,A}^{A} = -\,\underline{\alpha}_{B\,to\,A}^{A} \tag{3.5.2-26}$$

where

$\underline{\alpha}_{B\,to\,A}^{A}, \underline{\alpha}_{B\,to\,A}^{B}$ = Rotation error vector associated with the $\hat{C}_B^A$ matrix ("B to A") considering the A Frame to be misaligned, as projected onto A Frame, B Frame axes.

$\underline{\beta}_{B\,to\,A}^{A}, \underline{\beta}_{B\,to\,A}^{B}$ = Rotation error vector associated with the $\hat{C}_B^A$ matrix ("B to A") considering the B Frame to be misaligned, as projected onto A Frame, B Frame axes.

A helpful note to keep in mind when applying the above $\underline{\alpha}$, $\underline{\beta}$ definitions is that the $\underline{\alpha}$ misalignment considers the superscripted coordinate frame in the associated direction matrix (i.e., the A Frame in $\hat{C}_B^A$) to be the source of angular error, while the $\underline{\beta}$ misalignment considers the subscripted frame to be the angular error source (the B Frame in $\hat{C}_B^A$).

In summary, with (3.5.2-25), (3.5.2-26), (3.2.1-2) and (3.2.1-8), Equations (3.5.2-12), (3.5.2-13), (3.5.2-18), (3.5.2-19), (3.5.2-23), (3.5.2-24) and (3.5.2-21) become:

$$\left(\underline{\alpha}_{B\,to\,A}^{A}\times\right) = I - \hat{C}_B^A \left(C_B^A\right)^T \qquad \delta C_B^A = -\left(\underline{\alpha}_{B\,to\,A}^{A}\times\right) C_B^A \tag{3.5.2-27}$$

$$\left(\underline{\beta}_{B\,to\,A}^{B}\times\right) = \left(C_B^A\right)^T \hat{C}_B^A - I \qquad \delta C_B^A = C_B^A \left(\underline{\beta}_{B\,to\,A}^{B}\times\right) \tag{3.5.2-28}$$

$$\underline{\alpha}_{B\,to\,A}^{B} = \left(C_B^A\right)^T \underline{\alpha}_{B\,to\,A}^{A} \qquad \underline{\beta}_{B\,to\,A}^{A} = C_B^A \, \underline{\beta}_{B\,to\,A}^{B} \tag{3.5.2-29}$$

$$\left(\underline{\alpha}^{B}_{B\,to\,A}\times\right) = \left(C^{A}_{B}\right)^{T}\left(\underline{\alpha}^{A}_{B\,to\,A}\times\right)C^{A}_{B} \qquad \left(\underline{\beta}^{A}_{B\,to\,A}\times\right) = C^{A}_{B}\left(\underline{\beta}^{B}_{B\,to\,A}\times\right)\left(C^{A}_{B}\right)^{T} \tag{3.5.2-30}$$

$$\underline{\alpha}^{B}_{B\,to\,A} = -\,\underline{\beta}^{B}_{B\,to\,A} \qquad \underline{\beta}^{A}_{B\,to\,A} = -\,\underline{\alpha}^{A}_{B\,to\,A} \tag{3.5.2-31}$$

$$\left(\underline{\alpha}^{A}_{B\,to\,A}\times\right) = -\,C^{A}_{B}\left(\underline{\beta}^{B}_{B\,to\,A}\times\right)\left(C^{A}_{B}\right)^{T} \qquad \left(\underline{\beta}^{B}_{B\,to\,A}\times\right) = -\left(C^{A}_{B}\right)^{T}\left(\underline{\alpha}^{A}_{B\,to\,A}\times\right)C^{A}_{B} \tag{3.5.2-32}$$

A similar development can also be performed on the $\hat{C}^{B}_{A}$ matrix (which is the transpose of $\hat{C}^{A}_{B}$ because $\hat{C}^{A}_{B}$ is assumed in this section to be normal and orthogonal). We then can write:

$$\hat{C}^{B}_{A} = C^{B}_{A}\,C^{A}_{\hat{A}} = C^{B}_{A}\left[I + \left(\underline{\beta}^{A}_{A\,to\,B}\times\right)\right] \tag{3.5.2-33}$$

where

$\underline{\beta}^{A}_{A\,to\,B}$ = Rotation error vector associated with the $\hat{C}^{B}_{A}$ matrix ("A to B") considering Frame A to be misaligned, as projected on Frame A axes.

Upon rearrangement using (3.2.1-3) for the ideal C^{B}_{A} matrix, (3.5.2-33) becomes:

$$\left(\underline{\beta}^{A}_{A\,to\,B}\times\right) = \left(C^{B}_{A}\right)^{T}\hat{C}^{B}_{A} - I = C^{A}_{B}\left(\hat{C}^{A}_{B}\right)^{T} - I \tag{3.5.2-34}$$

Taking the transpose of (3.5.2-34) and recognizing that the transpose of a cross-product operator equals its negative, then yields:

$$\left(\underline{\beta}^{A}_{A\,to\,B}\times\right) = I - \hat{C}^{A}_{B}\left(C^{A}_{B}\right)^{T} \tag{3.5.2-35}$$

Comparing Equations (3.5.2-35) and (3.5.2-27) for $\left(\underline{\alpha}^{A}_{B\,to\,A}\times\right)$ shows that:

$$\underline{\beta}^{A}_{A\,to\,B} = \underline{\alpha}^{A}_{B\,to\,A} \tag{3.5.2-36}$$

Hence, the identical angle error vector is obtained by analyzing the error in $\hat{C}^{A}_{B}$ or $\hat{C}^{B}_{A}$ considering Frame A to be misaligned.

Equation (3.5.2-33) considers Frame A misalignment to be the source of the $\hat{C}^{B}_{A}$ error. A similar analysis can be performed considering B Frame misalignment as the $\hat{C}^{B}_{A}$ error source. Using (3.5.2-8) and (3.5.2-11) with B substituted for A and A substituted for B shows that:

$$\underline{\alpha}^{B}_{A\,to\,B} = \underline{\beta}^{B}_{B\,to\,A} \tag{3.5.2-37}$$

where

$\underline{\alpha}^{B}_{A\,\text{to}\,B}$ = Rotation error vector associated with the $\hat{C}^{B}_{A}$ matrix considering Frame B to be misaligned, as projected on Frame B axes.

Thus, the identical angle error vector is obtained by analyzing the error in $\hat{C}^{A}_{B}$ or $\hat{C}^{B}_{A}$ considering Frame B to be misaligned.

Finally, it is instructive to analyze the errors in a generalized C^{A}_{D} direction cosine matrix formed from the direction cosine matrix product chain rule:

$$C^{A}_{D} = C^{A}_{B}\, C^{B}_{D} \tag{3.5.2-38}$$

where

D = Arbitrary coordinate frame.

C^{B}_{D} = Direction cosine matrix that transforms vectors from the D to the B Frame.

C^{A}_{D} = Direction cosine matrix that transforms vectors from the D to the A Frame.

From Equations (3.5.2-27) we can write:

$$\left(\underline{\alpha}^{A}_{D\,\text{to}\,A}\times\right) = I - \hat{C}^{A}_{D}\left(C^{A}_{D}\right)^{T} \tag{3.5.2-39}$$

with

$$\hat{C}^{A}_{D} = \hat{C}^{A}_{B}\,\hat{C}^{B}_{D} \tag{3.5.2-40}$$

where

$\hat{C}^{A}_{D}, \hat{C}^{B}_{D}$ = C^{A}_{D}, C^{B}_{D} matrices calculated in the system, hence, containing errors.

$\underline{\alpha}^{A}_{D\,\text{to}\,A}$ = Rotation error vector associated with $\hat{C}^{A}_{D}$ ("D to A") considering the error in $\hat{C}^{A}_{D}$ to be produced by A Frame misalignment.

Applying (3.5.2-27) and general Equations (3.2.1-8) allows us to also write:

$$\left(\underline{\alpha}^{B}_{D\,\text{to}\,B}\times\right) = I - \hat{C}^{B}_{D}\left(C^{B}_{D}\right)^{T} \tag{3.5.2-41}$$

$$\left(\underline{\alpha}^{A}_{D\,\text{to}\,B}\times\right) = C^{A}_{B}\left(\underline{\alpha}^{B}_{D\,\text{to}\,B}\times\right)\left(C^{A}_{B}\right)^{T} \tag{3.5.2-42}$$

where

$\underline{\alpha}^{B}_{D\,to\,B}$ = Rotation error vector associated with $\hat{C}^{B}_{D}$ ("D to B") considering the error in $\hat{C}^{B}_{D}$ to be produced by B Frame misalignment.

Substituting (3.5.2-40) and (3.5.2-38) into (3.5.2-39), applying (3.5.2-41) for $\left(\underline{\alpha}^{B}_{D\,to\,B}\times\right)$, applying (3.5.2-27) for $\left(\underline{\alpha}^{A}_{B\,to\,A}\times\right)$ and dropping second order rotation error vector products then obtains:

$$\begin{aligned}
\left(\underline{\alpha}^{A}_{D\,to\,A}\times\right) &= I - \hat{C}^{A}_{B}\,\hat{C}^{B}_{D}\left(C^{B}_{D}\right)^{T}\left(C^{A}_{B}\right)^{T} \\
&= I - \hat{C}^{A}_{B}\,\hat{C}^{B}_{D}\left(C^{B}_{D}\right)^{T}\left(\hat{C}^{A}_{B}\right)^{-1}\hat{C}^{A}_{B}\left(C^{A}_{B}\right)^{T} \\
&= I - \hat{C}^{A}_{B}\left[I - \left(\underline{\alpha}^{B}_{D\,to\,B}\times\right)\right]\left(\hat{C}^{A}_{B}\right)^{-1}\left[I - \left(\underline{\alpha}^{A}_{B\,to\,A}\times\right)\right] \\
&= I - \left[\hat{C}^{A}_{B}\left(\hat{C}^{A}_{B}\right)^{-1} - \hat{C}^{A}_{B}\left(\underline{\alpha}^{B}_{D\,to\,B}\times\right)\left(\hat{C}^{A}_{B}\right)^{-1}\right]\left[I - \left(\underline{\alpha}^{A}_{B\,to\,A}\times\right)\right] \qquad (3.5.2\text{-}43) \\
&= I - \left[I - \hat{C}^{A}_{B}\left(\underline{\alpha}^{B}_{D\,to\,B}\times\right)\left(\hat{C}^{A}_{B}\right)^{-1}\right]\left[I - \left(\underline{\alpha}^{A}_{B\,to\,A}\times\right)\right] \\
&= \left(\underline{\alpha}^{A}_{B\,to\,A}\times\right) + \hat{C}^{A}_{B}\left(\underline{\alpha}^{B}_{D\,to\,B}\times\right)\left(\hat{C}^{A}_{B}\right)^{-1} - \hat{C}^{A}_{B}\left(\underline{\alpha}^{B}_{D\,to\,B}\times\right)\left(\hat{C}^{A}_{B}\right)^{-1}\left(\underline{\alpha}^{A}_{B\,to\,A}\times\right) \\
&\approx \left(\underline{\alpha}^{A}_{B\,to\,A}\times\right) + C^{A}_{B}\left(\underline{\alpha}^{B}_{D\,to\,B}\times\right)\left(C^{A}_{B}\right)^{T}
\end{aligned}$$

With (3.5.2-42), Equation (3.5.2-43) becomes:

$$\left(\underline{\alpha}^{A}_{D\,to\,A}\times\right) \approx \left(\underline{\alpha}^{A}_{D\,to\,B}\times\right) + \left(\underline{\alpha}^{A}_{B\,to\,A}\times\right) \qquad (3.5.2\text{-}44)$$

or equivalently,

$$\underline{\alpha}^{A}_{D\,to\,A} = \underline{\alpha}^{A}_{D\,to\,B} + \underline{\alpha}^{A}_{B\,to\,A} \qquad (3.5.2\text{-}45)$$

and with (3.5.2-29), (3.5.2-31), and the general form of Equation (3.5.2-23):

$$\underline{\beta}^{D}_{D\,to\,A} = \underline{\beta}^{D}_{D\,to\,B} + \underline{\beta}^{D}_{B\,to\,A} \qquad (3.5.2\text{-}46)$$

Equations (3.5.2-45) and (3.5.2-46) are the rotation error vector equivalents to Equation (3.5.2-40). They show that the rotation error vector for a direction cosine matrix formed as the product of two direction cosine matrices equals the sum of the rotation error vectors for the direction cosine matrices used in forming the product.

When applying Equations (3.5.2-45) and (3.5.2-46) it is helpful to recall the note following Equation (3.5.2-26) for the $\underline{\alpha}$ versus $\underline{\beta}$ definitions regarding the direction cosine matrices in Equation (3.5.2-40). Equation (3.5.2-45) considers superscripted coordinate frames in the Equation (3.5.2-40) direction cosine matrices to be the source of angular error (i.e., the A Frame in C_D^A, the A Frame in C_B^A and the B Frame in C_D^B). In contrast, Equation (3.5.2-46) considers subscripted coordinate frames in the Equation (3.5.2-40) direction cosine matrices to be the source of angular error (i.e., the D Frame in C_D^A, the B Frame in C_B^A and the D Frame in C_D^B).

For completeness, we summarize the analytical definitions for the angle error parameters in Equations (3.5.2-45) - (3.5.2-46) and their direction cosine matrix error equivalencies by applying generalized Equations (3.5.2-27) - (3.5.2-29) and (3.2.1-2) with (3.5.2-40):

$$\left(\underline{\alpha}_{D\,to\,B}^{B}\times\right) = I - \hat{C}_D^B\left(C_D^B\right)^T \qquad \delta C_D^B = -\left(\underline{\alpha}_{D\,to\,B}^{B}\times\right)C_D^B \qquad \underline{\alpha}_{D\,to\,B}^{A} = C_B^A\,\underline{\alpha}_{D\,to\,B}^{B}$$
$$\left(\underline{\alpha}_{B\,to\,A}^{A}\times\right) = I - \hat{C}_B^A\left(C_B^A\right)^T \qquad \delta C_B^A = -\left(\underline{\alpha}_{B\,to\,A}^{A}\times\right)C_B^A$$
$$\hat{C}_D^A = \hat{C}_B^A\,\hat{C}_D^B$$
$$\left(\underline{\alpha}_{D\,to\,A}^{A}\times\right) = I - \hat{C}_D^A\left(C_D^A\right)^T \qquad \delta C_D^A = -\left(\underline{\alpha}_{D\,to\,A}^{A}\times\right)C_D^A \tag{3.5.2-47}$$
$$\underline{\alpha}_{D\,to\,A}^{A} = \underline{\alpha}_{D\,to\,B}^{A} + \underline{\alpha}_{B\,to\,A}^{A}$$
$$\delta C_D^A = \delta\left(C_B^A\,C_D^B\right) = -\left[\left(\underline{\alpha}_{D\,to\,B}^{A} + \underline{\alpha}_{B\,to\,A}^{A}\right)\times\right]\left(C_B^A\,C_D^B\right)$$

$$\left(\underline{\beta}_{D\,to\,B}^{D}\times\right) = \left(C_D^B\right)^T\hat{C}_D^B - I \qquad \delta C_D^B = C_D^B\left(\underline{\beta}_{D\,to\,B}^{D}\times\right)$$
$$\left(\underline{\beta}_{B\,to\,A}^{B}\times\right) = \left(C_B^A\right)^T\hat{C}_B^A - I \qquad \delta C_B^A = C_B^A\left(\underline{\beta}_{B\,to\,A}^{B}\times\right) \qquad \underline{\beta}_{B\,to\,A}^{D} = \left(C_D^B\right)^T\underline{\beta}_{B\,to\,A}^{B}$$
$$\hat{C}_D^A = \hat{C}_B^A\,\hat{C}_D^B$$
$$\left(\underline{\beta}_{D\,to\,A}^{D}\times\right) = \left(C_D^A\right)^T\hat{C}_D^A - I \qquad \delta C_D^A = C_D^A\left(\underline{\beta}_{D\,to\,A}^{D}\times\right) \tag{3.5.2-48}$$
$$\underline{\beta}_{D\,to\,A}^{D} = \underline{\beta}_{D\,to\,B}^{D} + \underline{\beta}_{B\,to\,A}^{D}$$
$$\delta C_D^A = \delta\left(C_B^A\,C_D^B\right) = \left(C_B^A\,C_D^B\right)\left[\left(\underline{\beta}_{D\,to\,B}^{D} + \underline{\beta}_{B\,to\,A}^{D}\right)\times\right]$$

3.5.3 DIRECTION COSINE MATRIX MISALIGNMENT ERROR AS A FUNCTION OF EULER ANGLE ERRORS

The $\underline{\alpha}_{B\,to\,A}$, $\underline{\beta}_{B\,to\,A}$ rotation error vectors of Section 3.5.2 associated with the computed $\hat{C}_B^A$ direction cosine matrix (i.e., containing misalignment errors) can also be expressed in terms of errors in the Euler angles describing $\hat{C}_B^A$. Recall from Sections 3.2.3 and 3.2.3.1 that the error free C_B^A direction cosine matrix can be defined as in Equation (3.2.3.1-1) by:

$$C_B^A = C_{A_1}^A \, C_{A_2}^{A_1} \, C_B^{A_2} \tag{3.5.3-1}$$

where

A Frame = Starting arbitrary coordinate frame.

A_1 Frame = Frame A after rotating it about one of the A Frame axes through the "first" Euler angle rotation.

A_2 Frame = Frame A_1 after rotating it about one of the A_1 Frame axes through the "second" Euler angle rotation.

B Frame = Ending arbitrary coordinate frame obtained by rotating Frame A_2 about one of the A_2 Frame axes through the "third" Euler angle rotation.

C_B^A, $C_B^{A_2}$, $C_{A_2}^{A_1}$, $C_{A_1}^A$ = Direction cosine matrices that transform vectors from Frame B to A, from Frame B to A_2, from Frame A_2 to A_1 and from Frame A_1 to A.

In Section 3.2.3 and its subsections, the "first", "second" and "third" Euler angle rotations referenced in the above definitions were, respectively, heading about the A Frame Z axis, pitch about the A_1 Frame Y axis, and roll about the A_2 Frame X axis. In this section, we keep the development more general by not explicitly specifying the particular axes for the individual Euler rotations.

The system calculated form of Equation (3.5.3-1) (i.e., containing errors) is:

$$\hat{C}_B^A = \hat{C}_{A_1}^A \, \hat{C}_{A_2}^{A_1} \, \hat{C}_B^{A_2} \tag{3.5.3-2}$$

or equivalently:

$$\hat{C}_B^A = \hat{C}_{A_2}^A \, \hat{C}_B^{A_2} \qquad \hat{C}_{A_2}^A = \hat{C}_{A_1}^A \, \hat{C}_{A_2}^{A_1} \tag{3.5.3-3}$$

Applying general Equations (3.5.2-40) and (3.5.2-46) to Equations (3.5.3-3) and projecting onto Frame A axes shows that:

$$\underline{\beta}^{A}_{B\,to\,A} = \underline{\beta}^{A}_{A_2\,to\,A} + \underline{\beta}^{A}_{B\,to\,A_2} \qquad \underline{\beta}^{A}_{A_2\,to\,A} = \underline{\beta}^{A}_{A_1\,to\,A} + \underline{\beta}^{A}_{A_2\,to\,A_1} \tag{3.5.3-4}$$

and in combination:

$$\underline{\beta}^{A}_{B\,to\,A} = \underline{\beta}^{A}_{A_1\,to\,A} + \underline{\beta}^{A}_{A_2\,to\,A_1} + \underline{\beta}^{A}_{B\,to\,A_2} \tag{3.5.3-5}$$

where

$\underline{\beta}^{A}_{B\,to\,A}$ = Rotation error vector associated with $\hat{C}^{A}_{B}$ (considering the B Frame to be misaligned) projected on Frame A axes.

$\underline{\beta}^{A}_{A_1\,to\,A}$ = Rotation error vector associated with $\hat{C}^{A}_{A_1}$ (considering the A_1 Frame to be misaligned) projected on A Frame axes.

$\underline{\beta}^{A}_{A_2\,to\,A_1}$ = Rotation error vector associated with $\hat{C}^{A_1}_{A_2}$ (considering the A_2 Frame to be misaligned) projected on A Frame axes.

$\underline{\beta}^{A}_{B\,to\,A_2}$ = Rotation error vector associated with $\hat{C}^{A_2}_{B}$ (considering the B Frame to be misaligned) projected on A Frame axes.

Equation (3.5.3-5) is the rotation error vector equivalent to Equation (3.5.3-2). We now seek relationships between the terms on the right side of Equation (3.5.3-5) and the errors in the Euler angle parameters associated with $\hat{C}^{A}_{B}$. This is achieved by first rewriting Equation (3.5.3-5) in the following equivalent form using Equation (3.2.1-2) and the (3.2.1-5) chain rule:

$$\underline{\beta}^{A}_{B\,to\,A} = \underline{\beta}^{A}_{A_1\,to\,A} + C^{A}_{A_1}\,\underline{\beta}^{A_1}_{A_2\,to\,A_1} + C^{A}_{A_1}\,C^{A_1}_{A_2}\,\underline{\beta}^{A_2}_{B\,to\,A_2} \tag{3.5.3-6}$$

where

$\underline{\beta}^{A_1}_{A_2\,to\,A_1}$ = Rotation error vector associated with $\hat{C}^{A_1}_{A_2}$ (considering the A_2 Frame to be misaligned) projected on A_1 Frame axes.

$\underline{\beta}^{A_2}_{B\,to\,A_2}$ = Rotation error vector associated with $\hat{C}^{A_2}_{B}$ (considering the B Frame to be misaligned) projected on A_2 Frame axes.

Recall from Section 3.2.3 that each of the Euler angle parameters associated with C^{A}_{B} is defined as an angular rotation about a fixed axis so that in the error free case, the associated direction cosine matrix ($C^{A_2}_{B}$, $C^{A_1}_{A_2}$, or $C^{A}_{A_1}$) is given by the general Equations (3.2.3-2) form:

$$C = I + \sin\zeta\left(\underline{u}_\zeta\times\right) + (1 - \cos\,\zeta)\left(\underline{u}_\zeta\times\right)^2 \tag{3.5.3-7}$$

where

ζ = Generalized Euler rotation (e.g., in Section 3.2.3, ψ, θ or ϕ associated with C_B^A as in Equations (3.2.3-1)).

$\underline{u}_\zeta$ = Generalized rotation axis unit vector for the ζ Euler rotation (e.g., in Section 3.2.3, $\underline{u}_{ZA}^{A}$, $\underline{u}_{YA_1}^{A_1}$ or $\underline{u}_{XA_2}^{A_2}$ unit vectors along the A Frame Z-axis, A_1 Frame Y-axis or A_2 Frame X-axis as defined by Equations (3.2.3-3)).

C = Generalized direction cosine matrix associated with ζ and $\underline{u}_\zeta$ (e.g., in Section 3.2.3, $C_{A_1}^{A}$, $C_{A_2}^{A_1}$, or $C_B^{A_2}$).

Generalized Equations (3.5.2-28) and (3.5.2-30) show that:

$$\left(\underline{\beta}_{B\,to\,A}^{B}\times\right) = \left(C_B^A\right)^T \hat{C}_B^A - I \qquad \left(\underline{\beta}_{B\,to\,A}^{A}\times\right) = C_B^A \left(\underline{\beta}_{B\,to\,A}^{B}\times\right)\left(C_B^A\right)^T \tag{3.5.3-8}$$

or in combination:

$$\left(\underline{\beta}_{B\,to\,A}^{A}\times\right) = \hat{C}_B^A \left(C_B^A\right)^T - I \tag{3.5.3-9}$$

Generalizing Equation (3.5.3-9) to an arbitrary Euler angle rotation matrix gives:

$$\left(\underline{\beta}\times\right) = \hat{C}\, C^T - I \tag{3.5.3-10}$$

where

$\underline{\beta}$ = Generalized rotation error associated with $\hat{C}$ (e.g., $\hat{C}_{A_1}^{A}$, $\hat{C}_{A_2}^{A_1}$, or $\hat{C}_B^{A_2}$ as represented by $\underline{\beta}_{A_1\,to\,A}^{A}$, $\underline{\beta}_{A_2\,to\,A_1}^{A_1}$ or $\underline{\beta}_{B\,to\,A_2}^{A_2}$).

Returning to Equation (3.5.3-7) and defining the error in $\hat{C}$ to only be caused by error in the Euler angle ζ (i.e., not in $\underline{u}_\zeta$) allows us to write:

$$\hat{C} = I + \sin\hat{\zeta}\left(\underline{u}_\zeta\times\right) + (1 - \cos\hat{\zeta})\left(\underline{u}_\zeta\times\right)^2 \tag{3.5.3-11}$$

For simplicity we define:

$$\begin{aligned} f_1 &= \sin\zeta & f_2 &= 1 - \cos\zeta \\ \hat{f}_1 &= \sin\hat{\zeta} & \hat{f}_2 &= 1 - \cos\hat{\zeta} \end{aligned} \tag{3.5.3-12}$$

Substituting (3.5.3-7) and (3.5.3-11) with (3.5.3-12) into (3.5.3-10) then yields:

$$
\begin{aligned}
\left(\underline{\beta}\times\right) &= \hat{C}\,C^T - I = \left[I + \hat{f}_1\left(\underline{u}_\zeta\times\right) + \hat{f}_2\left(\underline{u}_\zeta\times\right)^2\right]\left[I + f_1\left(\underline{u}_\zeta\times\right) + f_2\left(\underline{u}_\zeta\times\right)^2\right]^T - I \\
&= \left[I + \hat{f}_1\left(\underline{u}_\zeta\times\right) + \hat{f}_2\left(\underline{u}_\zeta\times\right)^2\right]\left[I - f_1\left(\underline{u}_\zeta\times\right) + f_2\left(\underline{u}_\zeta\times\right)^2\right] - I \\
&= I + \left(\hat{f}_1 - f_1\right)\left(\underline{u}_\zeta\times\right) + \left(\hat{f}_2 + f_2 - \hat{f}_1\,f_1\right)\left(\underline{u}_\zeta\times\right)^2 + \left(\hat{f}_1\,f_2 - \hat{f}_2\,f_1\right)\left(\underline{u}_\zeta\times\right)^3 + \hat{f}_2\,f_2\left(\underline{u}_\zeta\times\right)^4 - I \\
&= \left(\hat{f}_1 - f_1\right)\left(\underline{u}_\zeta\times\right) + \left(\hat{f}_2 + f_2 - \hat{f}_1\,f_1\right)\left(\underline{u}_\zeta\times\right)^2 + \left(\hat{f}_1\,f_2 - \hat{f}_2\,f_1\right)\left(\underline{u}_\zeta\times\right)^3 + \hat{f}_2\,f_2\left(\underline{u}_\zeta\times\right)^4
\end{aligned}
\qquad (3.5.3\text{-}13)
$$

Alternative expressions for the $\left(\underline{u}_\zeta\times\right)^3$, $\left(\underline{u}_\zeta\times\right)^4$ terms in (3.5.3-13) can be derived by application of the (3.1.1-16) general vector triple cross product formula. First we find for arbitrary $\underline{V}_3$ using (3.1.1-15):

$$
\left(\underline{u}_\zeta\times\right)^2\underline{V}_3 = \underline{u}_\zeta\times\left(\underline{u}_\zeta\times\underline{V}_3\right) = \underline{u}_\zeta\left(\underline{u}_\zeta\cdot\underline{V}_3\right) - \underline{V}_3\left(\underline{u}_\zeta\cdot\underline{u}_\zeta\right) = -\underline{V}_3 + \underline{u}_\zeta\left(\underline{u}_\zeta\cdot\underline{V}_3\right) \qquad (3.5.3\text{-}14)
$$

Then, taking the cross-product of $\underline{u}_\zeta$ with (3.5.3-14)) we obtain:

$$
\left(\underline{u}_\zeta\times\right)^3\underline{V}_3 = \left(\underline{u}_\zeta\times\right)\left(\underline{u}_\zeta\times\right)^2\underline{V}_3 = \left(\underline{u}_\zeta\times\right)\left[-\underline{V}_3 + \underline{u}_\zeta\left(\underline{u}_\zeta\cdot\underline{V}_3\right)\right] = -\left(\underline{u}_\zeta\times\right)\underline{V}_3 \qquad (3.5.3\text{-}15)
$$

Lastly, the cross-product of $\underline{u}_\zeta$ with (3.5.3-15) gives:

$$
\left(\underline{u}_\zeta\times\right)^4\underline{V}_3 = -\left(\underline{u}_\zeta\times\right)^2\underline{V}_3 \qquad (3.5.3\text{-}16)
$$

Since $\underline{V}_3$ is arbitrary, (3.5.3-15) and (3.5.3-16) are equivalently:

$$
\left(\underline{u}_\zeta\times\right)^3 = -\left(\underline{u}_\zeta\times\right) \qquad \left(\underline{u}_\zeta\times\right)^4 = -\left(\underline{u}_\zeta\times\right)^2 \qquad (3.5.3\text{-}17)
$$

Substituting (3.5.3-17) into (3.5.3-13) obtains:

$$
\left(\underline{\beta}\times\right) = \left(\hat{f}_1 - f_1 - \hat{f}_1\,f_2 + \hat{f}_2\,f_1\right)\left(\underline{u}_\zeta\times\right) + \left(\hat{f}_2 + f_2 - \hat{f}_1\,f_1 - \hat{f}_2\,f_2\right)\left(\underline{u}_\zeta\times\right)^2 \qquad (3.5.3\text{-}18)
$$

With (3.5.3-12) the coefficient terms in (3.5.3-18) become:

$$
\begin{aligned}
\left(\hat{f}_1 - f_1 - \hat{f}_1\,f_2 + \hat{f}_2\,f_1\right) &= \sin\hat{\zeta} - \sin\zeta - \sin\hat{\zeta}\,(1 - \cos\zeta) + (1 - \cos\hat{\zeta})\sin\zeta \\
&= \sin\hat{\zeta}\cos\zeta - \cos\hat{\zeta}\sin\zeta = \sin(\hat{\zeta} - \zeta) \\
\left(\hat{f}_2 + f_2 - \hat{f}_1\,f_1 - \hat{f}_2\,f_2\right) &= (1 - \cos\hat{\zeta}) + (1 - \cos\zeta) - \sin\hat{\zeta}\sin\zeta - (1 - \cos\hat{\zeta})\,(1 - \cos\zeta) \\
&= 1 - \cos\hat{\zeta}\cos\zeta - \sin\hat{\zeta}\sin\zeta = 1 - \cos(\hat{\zeta} - \zeta)
\end{aligned}
\qquad (3.5.3\text{-}19)
$$

We define:

$$\delta\zeta \equiv \hat{\zeta} - \zeta \qquad (3.5.3\text{-}20)$$

where

$\delta\zeta$ = Error in the computed Euler angle $\hat{\zeta}$.

Substituting (3.5.3-19) with (3.5.3-20) into (3.5.3-18), and recognizing that $\delta\zeta$ is small, yields to first order in $\delta\zeta$:

$$\left(\underline{\beta}\times\right) = \delta\zeta\left(\underline{u}_{\zeta}\times\right) \qquad (3.5.3\text{-}21)$$

or

$$\underline{\beta} = \delta\zeta\, \underline{u}_{\zeta} \qquad (3.5.3\text{-}22)$$

Applying generalized Equation (3.5.3-22) obtains the desired relationship between the ($\underline{\beta}^{A}_{A_1 \text{to} A}$, $\underline{\beta}^{A_1}_{A_2 \text{to} A_1}$, $\underline{\beta}^{A_2}_{B \text{to} A_2}$) errors in ($\hat{C}^{A}_{A_1}$, $\hat{C}^{A_1}_{A_2}$, $\hat{C}^{A_2}_{B}$) for Equation (3.5.3-6), and the errors in the Euler angles associated with $\hat{C}^{A}_{B}$ around their defining axes. Repeating Equation (3.5.3-6) for completeness then finds:

$$\underline{\beta}^{A}_{B \text{to} A} = \underline{\beta}^{A}_{A_1 \text{to} A} + C^{A}_{A_1}\, \underline{\beta}^{A_1}_{A_2 \text{to} A_1} + C^{A}_{A_1}\, C^{A_1}_{A_2}\, \underline{\beta}^{A_2}_{B \text{to} A_2} \qquad (3.5.3\text{-}23)$$

$$\underline{\beta}^{A}_{A_1 \text{to} A} = \delta\zeta_1\, \underline{u}^{A}_{\zeta A} \qquad \underline{\beta}^{A_1}_{A_2 \text{to} A_1} = \delta\zeta_2\, \underline{u}^{A_1}_{\zeta A_1} \qquad \underline{\beta}^{A_2}_{B \text{to} A_2} = \delta\zeta_3\, \underline{u}^{A_2}_{\zeta A_2} \qquad (3.5.3\text{-}24)$$

with

$$\delta\zeta_1 \equiv \hat{\zeta}_1 - \zeta_1 \qquad \delta\zeta_2 \equiv \hat{\zeta}_2 - \zeta_2 \qquad \delta\zeta_3 \equiv \hat{\zeta}_3 - \zeta_3 \qquad (3.5.3\text{-}25)$$

where

$\underline{u}^{A}_{\zeta A}$, $\underline{u}^{A_1}_{\zeta A_1}$, $\underline{u}^{A_2}_{\zeta A_2}$ = Unit vectors along one of the Frame A, Frame A_1, Frame A_2 coordinate axes for the first, second, third Euler rotations in the Euler angle rotation sequence associated with C^{A}_{B}.

ζ_1, ζ_2, ζ_3 = Idealized error free values for the first, second, third Euler rotation angles around the $\underline{u}^{A}_{\zeta A}$, $\underline{u}^{A_1}_{\zeta A_1}$, $\underline{u}^{A_2}_{\zeta A_2}$ axes.

$\hat{\zeta}_1$, $\hat{\zeta}_2$, $\hat{\zeta}_3$ = System calculated values for ζ_1, ζ_2, ζ_3 from $\hat{C}^{A}_{B}$ (i.e., containing errors).

$\delta\zeta_1$, $\delta\zeta_2$, $\delta\zeta_3$ = The errors in $\hat{\zeta}_1$, $\hat{\zeta}_2$, $\hat{\zeta}_3$.

As an example, we note that for the particular Euler angle sequence described in Section 3.2.3, Equations (3.2.3-2) and (3.2.3-4):

$$\zeta_1 = \psi \qquad \zeta_2 = \theta \qquad \zeta_3 = \phi$$

$$\underline{u}^{A}_{\zeta A} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \underline{u}^{A_1}_{\zeta A_1} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \underline{u}^{A_2}_{\zeta A_2} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$C^{A}_{A_1} = \begin{bmatrix} \cos\zeta_1 & -\sin\zeta_1 & 0 \\ \sin\zeta_1 & \cos\zeta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad C^{A_1}_{A_2} = \begin{bmatrix} \cos\zeta_2 & 0 & \sin\zeta_2 \\ 0 & 1 & 0 \\ -\sin\zeta_2 & 0 & \cos\zeta_2 \end{bmatrix} \tag{3.5.3-26}$$

$$C^{A_2}_{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\zeta_3 & -\sin\zeta_3 \\ 0 & \sin\zeta_3 & \cos\zeta_3 \end{bmatrix}$$

where

ψ, θ, ϕ = Heading, pitch, roll Euler angles associated with C^{A}_{B} around the Frame A Z-axis, Frame A_1 Y-axis and Frame A_2 X-axis.

Equations (3.5.3-26) represent values for the generalized parameters developed in this section for only one of many possible Euler angle sequences. Other possible sequences can have different rotation axes for the Euler angle rotations; some can have two, four or more rotation axes, depending on the particular problem being analyzed.

Returning to the general problem, Equation (3.5.3-23) and (3.5.3-24) can also be written in the equivalent form:

$$\underline{\beta}^{A}_{B\,\mathrm{to}\,A} = \delta\zeta_1 \, \underline{u}^{A}_{\zeta A} + \delta\zeta_2 \, \underline{u}^{A}_{\zeta A_1} + \delta\zeta_3 \, \underline{u}^{A}_{\zeta A_2} \tag{3.5.3-27}$$

with

$$\underline{u}^{A}_{\zeta A_1} = C^{A}_{A_1} \, \underline{u}^{A_1}_{\zeta A_1} \qquad \underline{u}^{A}_{\zeta A_2} = C^{A}_{A_1} \, C^{A_1}_{A_2} \, \underline{u}^{A_2}_{\zeta A_2} \tag{3.5.3-28}$$

Equations (3.5.3-27) and (3.5.3-28) written in the B Frame are equivalently:

$$\underline{\beta}_{B\,to\,A}^{B} = \delta\zeta_1\,\underline{u}_{\zeta A}^{B} + \delta\zeta_2\,\underline{u}_{\zeta A_1}^{B} + \delta\zeta_3\,\underline{u}_{\zeta A_2}^{B}$$

$$\underline{u}_{\zeta A}^{B} = C_{A_2}^{B}\,C_{A_1}^{A_2}\,\underline{u}_{\zeta A}^{A_1} \qquad \underline{u}_{\zeta A_1}^{B} = C_{A_2}^{B}\,\underline{u}_{\zeta A_1}^{A_2} \tag{3.5.3-29}$$

Because each Euler rotation is a rotation vector, Equation (3.2.2.1-6) shows that the rotation axis components for a particular Euler angle are identical in the coordinate frames before and after the Euler rotation. Consequently:

$$\underline{u}_{\zeta A}^{A_1} = \underline{u}_{\zeta A}^{A} \qquad \underline{u}_{\zeta A_1}^{A_2} = \underline{u}_{\zeta A_1}^{A_1} \qquad \underline{u}_{\zeta A_2}^{B} = \underline{u}_{\zeta A_2}^{A_2} \tag{3.5.3-30}$$

With (3.5.3-30), Equations (3.5.3-29) become:

$$\underline{\beta}_{B\,to\,A}^{B} = \delta\zeta_1\,\underline{u}_{\zeta A}^{B} + \delta\zeta_2\,\underline{u}_{\zeta A_1}^{B} + \delta\zeta_3\,\underline{u}_{\zeta A_2}^{A_2} \tag{3.5.3-31}$$

with

$$\underline{u}_{\zeta A}^{B} = C_{A_2}^{B}\,C_{A_1}^{A_2}\,\underline{u}_{\zeta A}^{A} \qquad \underline{u}_{\zeta A_1}^{B} = C_{A_2}^{B}\,\underline{u}_{\zeta A_1}^{A_1} \tag{3.5.3-32}$$

From Equations (3.5.2-31), (3.5.3-5) and (3.5.3-23) we also see that:

$$\underline{\alpha}_{B\,to\,A}^{A} = \underline{\alpha}_{A_1\,to\,A}^{A} + C_{A_1}^{A}\,\underline{\alpha}_{A_2\,to\,A_1}^{A_1} + C_{A_1}^{A}\,C_{A_2}^{A_1}\,\underline{\alpha}_{B\,to\,A_2}^{A_2} \tag{3.5.3-33}$$

$$\underline{\alpha}_{A_1\,to\,A}^{A} = -\,\delta\zeta_1\,\underline{u}_{\zeta A}^{A} \qquad \underline{\alpha}_{A_2\,to\,A_1}^{A_1} = -\,\delta\zeta_2\,\underline{u}_{\zeta A_1}^{A_1} \qquad \underline{\alpha}_{B\,to\,A_2}^{A_2} = -\,\delta\zeta_3\,\underline{u}_{\zeta A_2}^{A_2} \tag{3.5.3-34}$$

where

$\underline{\alpha}_{B\,to\,A}^{A}$ = Rotation error vector associated with $\hat{C}_{B}^{A}$ (considering the A Frame to be misaligned) projected on Frame A axes.

$\underline{\alpha}_{A_1\,to\,A}^{A}$ = Rotation error vector associated with $\hat{C}_{A_1}^{A}$ (considering the A Frame to be misaligned) projected on Frame A axes.

$\underline{\alpha}_{A_2\,to\,A_1}^{A_1}$ = Rotation error vector associated with $\hat{C}_{A_2}^{A_1}$ (considering the A_1 Frame to be misaligned) projected on Frame A_1 axes.

$\underline{\alpha}_{B\,to\,A_2}^{A_2}$ = Rotation error vector associated with $\hat{C}_{B}^{A_2}$ (considering the A_2 Frame to be misaligned) projected on Frame A_2 axes.

or alternatively:

$$\underline{\alpha}_{B\,to\,A}^{A} = -\,\delta\zeta_1\,\underline{u}_{\zeta A}^{A} - \delta\zeta_2\,\underline{u}_{\zeta A_1}^{A} - \delta\zeta_3\,\underline{u}_{\zeta A_2}^{A} \tag{3.5.3-35}$$

with $\underline{u}_{\zeta A_1}^{A}$ and $\underline{u}_{\zeta A_2}^{A}$ defined in Equations (3.5.3-28).

In the B Frame, from Equations (3.5.2-31) and (3.5.3-31) we also have:

$$\underline{\alpha}_{B\,to\,A}^{B} = -\,\delta\zeta_1\,\underline{u}_{\zeta A}^{B} - \delta\zeta_2\,\underline{u}_{\zeta A_1}^{B} - \delta\zeta_3\,\underline{u}_{\zeta A_2}^{A_2} \tag{3.5.3-36}$$

with $\underline{u}_{\zeta A}^{B}$ and $\underline{u}_{\zeta A_1}^{B}$ defined by Equations (3.5.3-32).

Finally, it is instructive to note that the form of Equations (3.5.3-23) with (3.5.3-24), Equations (3.5.3-31) with (3.5.3-32), Equations (3.5.3-33) with (3.5.3-34) and Equations (3.5.3-36) with (3.5.3-32) lend themselves to the Section 3.2.3.3 "Method of Least Work" diagram technique for analyzing Euler angle defined attitude error characteristics. For example, for the particular Euler sequence described in Section 3.2.3 (as defined in this section by Equations (3.5.3-26)) the former equations can be represented by such diagrams as in Figures 3.5.3-1 and 3.5.3-2:

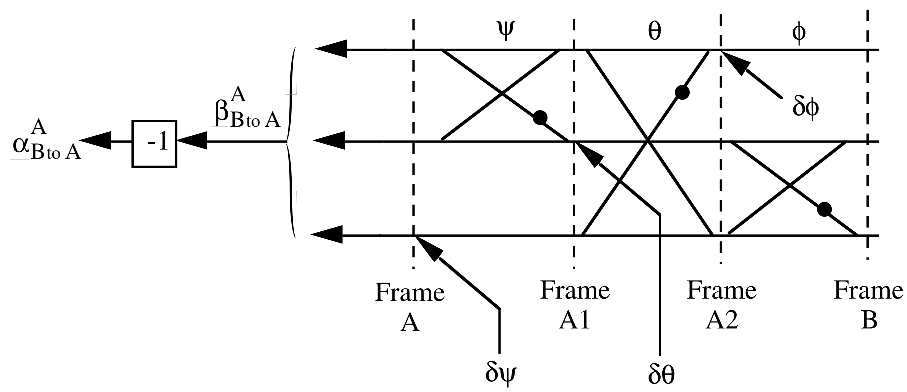


Figure 3.5.3-1 Orientation Error Vectors In The A Frame As A Function Of Euler Angle Errors

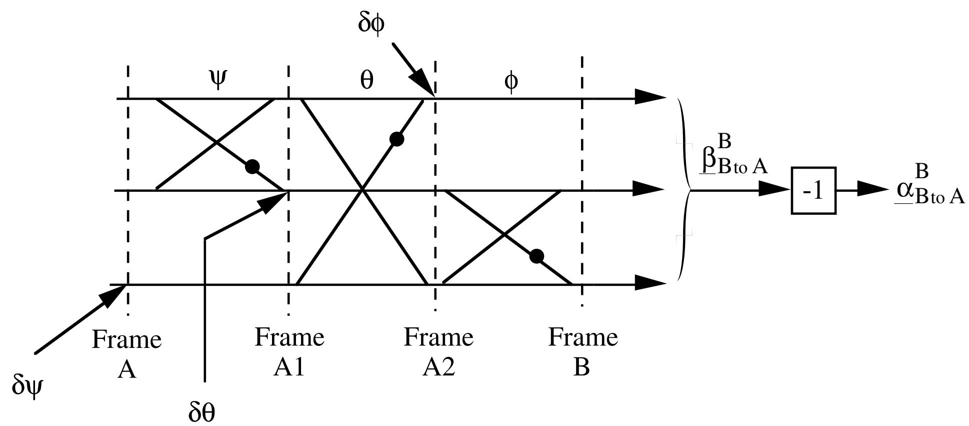


Figure 3.5.3-2 Orientation Error Vectors In The B Frame As A Function Of Euler Angle Errors

3.5.4 VECTOR ERROR CHARACTERISTICS

Consider an arbitrary vector with an ideal error free value and a system computed value containing errors. We define the error in the vector in terms of the coordinate frame in which it is evaluated and the coordinate frame in which the evaluated vector error is projected:

$$\delta\underline{V}_{B}^{A} \equiv C_{B}^{A}\left(\hat{\underline{V}}^{B} - \underline{V}^{B}\right) \qquad (3.5.4\text{-}1)$$

where

A, B = Arbitrary coordinate frames.

C_{B}^{A} = Direction cosine matrix that transforms vectors from the B Frame to the A Frame.

$\underline{V}^{B}$ = Arbitrary idealized error free vector projected on coordinate Frame B axes.

$\hat{\underline{V}}^{B}$ = System computed value of $\underline{V}^{B}$ (i.e., containing errors) projected on B Frame axes.

$\delta\underline{V}_{B}^{A}$ = Error in $\hat{\underline{V}}$ evaluated in the B Frame (subscript label) and then projected on A Frame axes (superscript label).

From the (3.5.4-1) definition, we can also project the B Frame evaluated error vector onto B Frame axes to obtain:

$$\delta \underline{V}_B^B = \hat{\underline{V}}^B - \underline{V}^B \tag{3.5.4-2}$$

Alternatively, we can evaluate the $\underline{V}$ error in the A Frame and project it onto A Frame axes as:

$$\delta \underline{V}_A^A = \hat{\underline{V}}^A - \underline{V}^A \tag{3.5.4-3}$$

where

$\delta \underline{V}_A^A$ = Error in $\hat{\underline{V}}$ evaluated in the A Frame (subscript label) and then projected on A Frame axes (superscript label).

But,

$$\hat{\underline{V}}^A = \hat{C}_B^A \, \hat{\underline{V}}^B \qquad \underline{V}^A = C_B^A \, \underline{V}^B \tag{3.5.4-4}$$

where

$\hat{C}_B^A$ = System computed value of C_B^A (i.e., containing errors).

Following the development approach in Section 3.5.2 (Equations (3.5.2-1) and (3.5.2-27)) we can write:

$$\hat{C}_B^A = C_B^A + \delta C_B^A = C_B^A - \left(\underline{\alpha}_{B\,to\,A}^A \times\right) C_B^A \tag{3.5.4-5}$$

where

$\underline{\alpha}_{B\,to\,A}^A$ = Rotation error vector associated with the $\hat{C}_B^A$ matrix (considering the $\hat{C}_B^A$ error to be A Frame misalignment) as projected onto A Frame axes.

Substituting (3.5.4-4) and (3.5.4-5) into (3.5.4-3) and applying (3.5.4-1) and (3.5.4-2) then yields:

$$\begin{aligned}
\delta \underline{V}_A^A &= \left[C_B^A - \left(\underline{\alpha}_{B\,to\,A}^A \times\right) C_B^A\right] \hat{\underline{V}}^B - \underline{V}^A \\
&= \left[C_B^A - \left(\underline{\alpha}_{B\,to\,A}^A \times\right) C_B^A\right] \left(\underline{V}^B + \delta \underline{V}_B^B\right) - \underline{V}^A \\
&= C_B^A \, \underline{V}^B + C_B^A \, \delta \underline{V}_B^B - \left(\underline{\alpha}_{B\,to\,A}^A \times\right) C_B^A \, \underline{V}^B - \left(\underline{\alpha}_{B\,to\,A}^A \times\right) C_B^A \, \delta \underline{V}_B^B - \underline{V}^A \\
&= \underline{V}^A + \delta \underline{V}_B^A - \left(\underline{\alpha}_{B\,to\,A}^A \times\right) \underline{V}^A - \left(\underline{\alpha}_{B\,to\,A}^A \times\right) \delta \underline{V}_B^A - \underline{V}^A \\
&= \delta \underline{V}_B^A - \underline{\alpha}_{B\,to\,A}^A \times \underline{V}^A - \underline{\alpha}_{B\,to\,A}^A \times \delta \underline{V}_B^A
\end{aligned} \tag{3.5.4-6}$$

or, after rearrangement and neglecting the $\underline{\alpha}_{B\,to\,A}^A \times \delta \underline{V}_B^A$ term as second order:

$$\delta\underline{V}_B^A = \delta\underline{V}_A^A + \underline{\alpha}_{B\,to\,A}^A \times \underline{V}^A \tag{3.5.4-7}$$

Transforming Equation (3.5.4-7) to the B Frame also yields after rearrangement:

$$\delta\underline{V}_A^B = \delta\underline{V}_B^B - \underline{\alpha}_{B\,to\,A}^B \times \underline{V}^B \tag{3.5.4-8}$$

where

$\underline{\alpha}_{B\,to\,A}^B$ = Rotation error vector associated with the $\hat{C}_B^A$ matrix (considering the $\hat{C}_B^A$ error to be A Frame misalignment) as projected onto B Frame axes.

Finally, we can substitute (3.5.2-31) into Equations (3.5.4-7) and (3.5.4-8) to obtain the equivalent forms:

$$\delta\underline{V}_B^A = \delta\underline{V}_A^A - \underline{\beta}_{B\,to\,A}^A \times \underline{V}^A \tag{3.5.4-9}$$

$$\delta\underline{V}_A^B = \delta\underline{V}_B^B + \underline{\beta}_{B\,to\,A}^B \times \underline{V}^B \tag{3.5.4-10}$$

where

$\underline{\beta}_{B\,to\,A}^A, \underline{\beta}_{B\,to\,A}^B$ = Rotation error vector associated with the $\hat{C}_B^A$ matrix (considering the $\hat{C}_B^A$ error to be B Frame misalignment) as projected onto A Frame, B Frame axes.

4 Continuous Form Strapdown Inertial Navigation Equations

4.0 OVERVIEW

A strapdown inertial navigation system consists of a computer and an inertial sensor assembly typically containing three orthogonal strapdown accelerometers and angular rate sensors. The sensor assembly data is input to the computer which performs five basic operations:

- Computation of sensor assembly angular attitude orientation using an integration algorithm operating on the angular rate sensor input data.

- Applying the computed attitude to transform input strapdown accelerometer signals from sensor axes to navigation coordinates.

- Processing the transformed acceleration data with an integration algorithm to calculate velocity relative to the earth in the navigation frame (including the addition of gravitational acceleration effects not measured by the accelerometers).

- Processing the computed velocity with an integration algorithm to calculate position location in a position reference coordinate system.

- Converting the attitude, velocity and position data to equivalent forms required for system output.

Chapters 7 and 19 (Section 19.1) provide in-depth derivations of digital computation algorithms for attitude/velocity/position integration and acceleration transformation. In this chapter, we will derive an equivalent set of equations for the above operations assuming an idealized infinitely fast system computer with continuous sensor assembly angular rate/acceleration inputs. The integration operations will be developed in the form of continuous differential equations that when integrated in the classical analytical continuous sense, would provide the attitude, velocity and position data generated digitally in the strapdown system computer. As such, the differential equations derived in this chapter can be viewed as the theoretical design goal for the Chapter 7 and 19 (Section 19.1) strapdown system computer digital algorithms. The Chapter 7 and 19 (Section 19.1) algorithms are designed to achieve the same numerical result by digital integration as the continuous integration of the differential equations developed in this chapter.

For a terrestrial (earth) based inertial navigation system, sensor assembly angular attitude orientation is usually described as an "attitude direction cosine matrix" (or attitude quaternion) relating sensor assembly axes (the "body" or B Frame) to locally level attitude reference coordinates (L Frame). Section 4.1 develops the differential equations for the attitude direction cosine matrix and attitude quaternion rate as a function of input strapdown angular rate sensor data. A correction term is included in the equations (Section 4.1.1) for angular rotation of the locally level reference L Frame to remain level (to account for earth's rotation rate and angular rotation of the local level relative to the earth due to navigation system horizontal velocity over the earth). Calculation of the correction term requires knowledge of the direction cosine matrix between a locally level navigation reference N Frame and an earth fixed reference coordinate E Frame. The N to E direction cosine matrix also defines the system angular position location in earth reference coordinates, hence, is sometimes denoted as the "position" direction cosine matrix. The position direction cosine matrix is calculated by integrating its differential equation (derived in Section 4.4.1.1) using system computed velocity relative to the earth as input. Section 4.1.2 describes how the attitude direction cosine matrix is also used to compute roll, pitch, heading Euler angle outputs.

Sections 4.2 and 4.3 develop the velocity rate differential equation using transformed strapdown acceleration data as input. The velocity components are computed by integrating the differential equation in the locally level navigation coordinate N Frame. Section 4.3.1 describes how the N Frame velocity components can be converted to their equivalent components along local north, east, vertical axes (or alternatively, along earth referenced E Frame axes) for output.

Section 4.4 develops equations for determining position location relative to the earth. Sections 4.4.1.1 and 4.4.1.2 show how position in the form of the N to E position direction cosine matrix and altitude is calculated by an integration operation on N Frame velocity. A divergent error characteristic can develop in the vertical velocity/altitude integration process excited principally from accelerometer error input to the altitude rate equation. Section 4.4.1.2.1 discusses control correction terms in the vertical velocity/altitude integration operations that can be used to reduce vertical velocity/altitude error and prevent vertical channel divergence. Section 4.4.2 develops several analytical equivalencies for converting earth referenced position location data from one representation form to another (e.g., position direction matrix to latitude/longitude). Section 4.4.3 derives differential equations as a function of system computed velocity, that when integrated, provide latitude and longitude position location data directly. A discussion is included regarding the singularities present in this integration approach for position determination, in comparison with the Section 4.4.1.1 position direction cosine rate integration approach which is singularity free.

Section 4.5 provides a general discussion of various options that can be considered for navigation N Frame azimuth (heading) orientation selection. Section 4.6 discusses the general initialization requirements for the attitude, velocity and position integration functions in a strapdown INS, including various methods of implementation.

Section 4.7 provides a tabular reference summary of the salient equations derived in Chapter 4, listed in the order they would be processed in a strapdown INS computer.

The coordinate frames used in this chapter are the B, L, N, Geo, E and I Frames as defined in Section 2.2, and for more specificity, we expand the earth fixed E Frame definition to define a particular X, Y, Z axis arrangement where

E = Earth fixed coordinate frame with Y along the earth polar axis, Z, X in the earth equatorial plane, with Z in the Greenwich England meridian plane.

4.1 ATTITUDE RATE EQUATIONS

The attitude of the B Frame relative to the local level attitude reference Frame L is typically computed in a strapdown INS, either in the form of a direction cosine matrix or an attitude quaternion. Euler angles are not generally used as the basic attitude reference form due to their singularities and more complex attitude rate form (compare Euler angle rate Equations (3.3.3.2-4) with direction cosine rate Equations (3.3.2-15) and quaternion rate Equations (3.3.4-26)). Attitude determination consists of integrating the associated attitude rate equations for the selected attitude parameters. Thus, applying Equations (3.3.2-13) or (3.3.4-24) and the Section 2.2 coordinate frame definitions, attitude is typically computed by integration of:

$$\dot{C}_B^L = C_B^L \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IL}^L \times\right) C_B^L \tag{4.1-1}$$

or

$$\dot{q}_B^L = \frac{1}{2}\, q_B^L\, \omega_{IB}^B - \frac{1}{2}\, \omega_{IL}^L\, q_B^L \tag{4.1-2}$$

where

C_B^L, q_B^L = Direction cosine matrix and attitude quaternion relating coordinate Frames B and L.

$\underline{\omega}_{IB}^B$ = Angular rate of the B Frame relative to inertial space expressed in B Frame axes.

$\underline{\omega}_{IL}^L$ = Angular rate of the L Frame relative to inertial space expressed in L Frame coordinates.

ω_{IB}^B, ω_{IL}^L = Quaternion equivalents to $\underline{\omega}_{IB}^B$, $\underline{\omega}_{IL}^L$ as indicated below using (3.2.4.1-3).

$$\omega_{IB}^B \equiv \begin{bmatrix} 0 \\ \underline{\omega}_{IB}^B \end{bmatrix} \qquad \omega_{IL}^L \equiv \begin{bmatrix} 0 \\ \underline{\omega}_{IL}^L \end{bmatrix} \tag{4.1-3}$$

The $\underline{\omega}_{IB}^{B}$ components would be formed from measurements taken from the strapdown inertial angular rate sensors. The $\underline{\omega}_{IL}^{L}$ components would be formed from the definition of the L Frame and its respective angular rate (to be discussed subsequently).

The component forms of (4.1-1) and (4.1-2) are, from Equations (3.3.2-15) and (3.3.4-26):

$$
\begin{aligned}
\dot{C}_{11} &= C_{12}\,\omega_{ZB} - C_{13}\,\omega_{YB} + C_{21}\,\omega_{ZL} - C_{31}\,\omega_{YL} \\
\dot{C}_{12} &= C_{13}\,\omega_{XB} - C_{11}\,\omega_{ZB} + C_{22}\,\omega_{ZL} - C_{32}\,\omega_{YL} \\
\dot{C}_{13} &= C_{11}\,\omega_{YB} - C_{12}\,\omega_{XB} + C_{23}\,\omega_{ZL} - C_{33}\,\omega_{YL} \\[6pt]
\dot{C}_{21} &= C_{22}\,\omega_{ZB} - C_{23}\,\omega_{YB} + C_{31}\,\omega_{XL} - C_{11}\,\omega_{ZL} \\
\dot{C}_{22} &= C_{23}\,\omega_{XB} - C_{21}\,\omega_{ZB} + C_{32}\,\omega_{XL} - C_{12}\,\omega_{ZL} \\
\dot{C}_{23} &= C_{21}\,\omega_{YB} - C_{22}\,\omega_{XB} + C_{33}\,\omega_{XL} - C_{13}\,\omega_{ZL} \\[6pt]
\dot{C}_{31} &= C_{32}\,\omega_{ZB} - C_{33}\,\omega_{YB} + C_{11}\,\omega_{YL} - C_{21}\,\omega_{XL} \\
\dot{C}_{32} &= C_{33}\,\omega_{XB} - C_{31}\,\omega_{ZB} + C_{12}\,\omega_{YL} - C_{22}\,\omega_{XL} \\
\dot{C}_{33} &= C_{31}\,\omega_{YB} - C_{32}\,\omega_{XB} + C_{13}\,\omega_{YL} - C_{23}\,\omega_{XL}
\end{aligned}
\tag{4.1-4}
$$

$$
\begin{aligned}
\dot{a} &= \frac{1}{2}\left(- b\,\omega_{XB} - c\,\omega_{YB} - d\,\omega_{ZB} + b\,\omega_{XL} + c\,\omega_{YL} + d\,\omega_{ZL}\right) \\
\dot{b} &= \frac{1}{2}\left(a\,\omega_{XB} - d\,\omega_{YB} + c\,\omega_{ZB} - a\,\omega_{XL} - d\,\omega_{YL} + c\,\omega_{ZL}\right) \\
\dot{c} &= \frac{1}{2}\left(d\,\omega_{XB} + a\,\omega_{YB} - b\,\omega_{ZB} + d\,\omega_{XL} - a\,\omega_{YL} - b\,\omega_{ZL}\right) \\
\dot{d} &= \frac{1}{2}\left(- c\,\omega_{XB} + b\,\omega_{YB} + a\,\omega_{ZB} - c\,\omega_{XL} + b\,\omega_{YL} - a\,\omega_{ZL}\right)
\end{aligned}
\tag{4.1-5}
$$

with

$$
\underline{\omega}_{IB}^{B} \equiv \begin{bmatrix} \omega_{XB} \\ \omega_{YB} \\ \omega_{ZB} \end{bmatrix} \qquad
\underline{\omega}_{IL}^{L} \equiv \begin{bmatrix} \omega_{XL} \\ \omega_{YL} \\ \omega_{ZL} \end{bmatrix}
$$

$$
C_B^L \equiv \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \qquad
q_B^L \equiv \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}
\tag{4.1-6}
$$

As an option to integrating direction cosine rate equations for each of the three rows of C_B^L, the first two rows can be calculated by integration of the first six expressions in (4.1-4), with the third row then calculated from the first two based on the following expression taken from application of Equations (3.2.1-3) and (3.2.1-6):

$$C_L^B = \begin{bmatrix} \underline{u}_{1L}^B & \underline{u}_{2L}^B & \underline{u}_{3L}^B \end{bmatrix} \qquad C_B^L = \left(C_L^B\right)^T \tag{4.1-7}$$

where

$\underline{u}_{1L}^B, \underline{u}_{2L}^B, \underline{u}_{3L}^B$ = Unit vectors along the L Frame X, Y, Z axes (i.e., 1, 2, 3) projected on B Frame axes, which from (4.1-7), correspond to the rows of C_B^L.

Hence, from the definition of the (4.1-7) unit vectors we can write:

$$\underline{u}_{3L}^B = \underline{u}_{1L}^B \times \underline{u}_{2L}^B \tag{4.1-8}$$

Equation (4.1-8) can be utilized to calculate the third row of C_B^L from the first two rows. Using (4.1-6) and (4.1-7), Equation (4.1-8) in component form is:

$$\begin{aligned} C_{31} &= C_{12}\, C_{23} - C_{13}\, C_{22} \\ C_{32} &= C_{13}\, C_{21} - C_{11}\, C_{23} \\ C_{33} &= C_{11}\, C_{22} - C_{12}\, C_{21} \end{aligned} \tag{4.1-9}$$

4.1.1 ANGULAR RATE OF LOCAL LEVEL FRAME L

The inertial angular rate of the local level frame ($\underline{\omega}_{IL}^L$) in Equations (4.1-1) - (4.1-3) is the sum in Frame N coordinates of the angular rates of Frame L relative to Frame N, Frame N relative to Frame E and Frame E relative to Frame I, with the sum then transformed to Frame L coordinates. From the Section 2.2 definition of Frames L and N (they are fixed relative to one another), the relative angular rate between the two is zero, hence:

$$\underline{\omega}_{IL}^L = C_N^L \left(\underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N\right) \tag{4.1.1-1}$$

where

$\underline{\omega}_{IE}^N$ = Angular rate of the earth E Frame relative to inertial space as projected along N Frame axes.

$\underline{\omega}_{EN}^{N}$ = Angular rate of the navigation N Frame relative to the E Frame as projected along N Frame axes.

C_N^L = Direction cosine matrix that transforms vectors from the N to the L Frame.

with, by the Section 2.2 definition of the L and N Frames:

$$C_N^L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{4.1.1-2}$$

The $\underline{\omega}_{IE}^{N}$ term in (4.1.1-1) represents the earth rotation rate vector as seen in local level navigation coordinates. Using generalized Equations (3.2.1-2) and (3.2.1-3), $\underline{\omega}_{IE}^{N}$ is related to the equivalent component vector in earth coordinates through C_N^E, the direction cosine matrix relating local level Frame N and earth coordinate Frame E:

$$\underline{\omega}_{IE}^{N} = \left(C_N^E\right)^T \underline{\omega}_{IE}^{E} \tag{4.1.1-3}$$

with from the Section 4.0 definition for Frame E:

$$\underline{\omega}_{IE}^{E} = \begin{bmatrix} 0 & \omega_e & 0 \end{bmatrix}^T \tag{4.1.1-4}$$

where

ω_e = Earth angular rotation rate magnitude relative to inertial space.

The transpose (of a row) format in (4.1.1-4) is used merely to reduce the vertical space that would be required for the normal column matrix format. This approach will be used when convenient throughout the book.

The $\underline{\omega}_{EN}^{N}$ term in (4.1.1-1) is equal to the sum of its horizontal and vertical components. The vertical component is a function of the type of local level navigation frame utilized (e.g., wander azimuth, free azimuth, or North/East geographic, to be discussed in Section 4.5). The horizontal component of $\underline{\omega}_{EN}^{N}$ is the angular rate that will maintain the N Frame locally level under horizontal position movement over the earth (i.e., horizontal velocity). This is identical to the horizontal component of the angular rate of the horizontal velocity vector. In other words, if the N Frame is rotated at the same horizontal angular rate as the horizontal velocity vector, it will also remain horizontal (as does the horizontal velocity vector by definition). For a spherical earth, the magnitude of the horizontal velocity vector angular rate and the $\underline{\omega}_{EN}^{N}$ horizontal component (call it $\underline{\omega}_{EN_H}^N$), equals the horizontal component of velocity (call it $\underline{v}_H^N$) divided by the distance from earth's center to the inertial navigation system (call it R). The direction of $\underline{\omega}_{EN_H}^N$ is found using the "right hand rule" by curling the right hand fingers in the direction of $\underline{v}_H^N$ over the surface of the earth (you need a large hand for this) which then places the thumb along $\underline{\omega}_{EN_H}^N$. Equivalently, $\underline{\omega}_{EN_H}^N$ can be calculated as the cross-product between an upward unit vector and $\underline{v}_H^N$ divided by R, or since the upward velocity component has no component perpendicular to the upward vector, as the cross-product between the upward unit vector and the total velocity vector divided by R. The previous effects can be expressed analytically as:

$$\underline{\rho}^N \equiv \underline{\omega}_{EN}^N = \frac{1}{R}\left(\underline{u}_R^N \times \underline{v}^N\right) + \rho_{ZN}\,\underline{u}_R^N \tag{4.1.1-5}$$

where

$\underline{\rho}^N$ = Conventional notation for $\underline{\omega}_{EN}^N$, also known as "transport rate", and analytically defined as the angular rate of Frame N relative to Frame E.

$\underline{v}^N$ = Translational velocity of the navigation frame relative to the earth.

R = Distance from earth's center to the inertial navigation system.

$\underline{u}_R^N$ = Unit vector along the position vector from earth center to the current position.

ρ_{ZN} = Vertical component of $\underline{\rho}^N$. Options for calculating ρ_{ZN} are discussed in Section 4.5.

For the conventional ellipsoidal earth surface model (see Section 5.1), Equation (4.1.1-5) has the more general form:

$$\underline{\rho}^N \equiv \underline{\omega}_{EN}^N = F_C^N\left(\underline{u}_{ZN}^N \times \underline{v}^N\right) + \rho_{ZN}\,\underline{u}_{ZN}^N \tag{4.1.1-6}$$

where

F_C^N = Curvature matrix in the N Frame (3 by 3) that is a function of position location over the earth, with elements 3, i and i, 3 equal to zero and the remaining elements symmetrical about the diagonal. For a spherical earth model, the "remaining" elements are zero off the diagonal and the reciprocal of the radial distance from earth center to the INS on the diagonal. For an ellipsoidal earth surface shape model, the "remaining" terms represent the local curvature on the earth surface projected to the INS altitude (See Equations (5.3-18) for closed-form expression).

$\underline{u}_{ZN}^{N}$ = Unit vector perpendicular to the surface of the earth, upward along the local vertical (i.e., along the Z axis of the N Frame as defined in Section 2.2).

The C_N^E direction cosine matrix is required to transform earth rate from Frame E to Frame N coordinates in (4.1.1-3), and to compute F_C^N, ρ_{ZN} in (4.1.1-6). The C_N^E matrix is also used (as described in Section 4.4) to describe horizontal position location over the earth. The C_N^E matrix is computed by an integration operation on $\underline{\rho}^N$ transport rate as part of the position determination process described in Section 4.4.1.1.

The component form of Equation (4.1.1-1) for Equations (4.1-4) and (4.1-5) is obtained by substitution of (4.1.1-3) and (4.1.1-6) for $\underline{\omega}_{IE}^{N}$ and $\underline{\omega}_{EN}^{N}$, and insertion of the appropriate matrix and vector definitions from Equations (4.1.1-2), (4.1.1-4) and (4.4.1.1-2):

$$
\begin{aligned}
\omega_{XL} &= D_{22}\,\omega_e + \rho_{YN} \\
\omega_{YL} &= D_{21}\,\omega_e + \rho_{XN} \\
\omega_{ZL} &= -\,D_{23}\,\omega_e - \rho_{ZN}
\end{aligned}
\qquad (4.1.1\text{-}7)
$$

4.1.2 EULER ANGLE OUTPUTS

Although not explicitly calculated by integration, Euler angle data is still typically required from an inertial navigation system for output information. For the case when the basic attitude parameters (computed by integration) are direction cosines, the Euler angles can be extracted by application of Equations (3.2.3.2-1), (3.2.3.2-2) and (3.2.3.2-4):

$$\theta = \tan^{-1} \frac{-\,C_{31}}{\sqrt{1 - C_{31}{}^2}}$$

For $\left| C_{31} \right| < 0.999$: (4.1.2-1)

$$\phi = \tan^{-1} \frac{C_{32}}{C_{33}} \qquad \psi = \tan^{-1} \frac{C_{21}}{C_{11}}$$

(Continued)

For $C_{31} \leq -0.999$:

$$\psi - \phi = \tan^{-1} \frac{C_{23} - C_{12}}{C_{13} + C_{22}} \qquad \text{(4.1.2-1) (Continued)}$$

For $C_{31} \geq 0.999$

$$\psi + \phi = \pi + \tan^{-1} \frac{C_{23} + C_{12}}{C_{13} - C_{22}}$$

The previous equations define the Euler angles relating the L and B Frames. For the situation when $|C_{31}| \geq 0.999$, the equations shown can be utilized for ψ, ϕ, with either ψ or ϕ specified based on some other criteria (e.g., the value when the $|C_{31}| \geq 0.999$ region was entered). Alternatively, both ψ and ϕ can be frozen at their values when the $|C_{31}| \geq 0.999$ region was entered.

For the situation when the attitude quaternion is the basic attitude parameter (computed by integration), there is no direct way, to the author's knowledge, for calculating the equivalent Euler angle parameters directly from the quaternion component elements. The method to be used, therefore, is to apply Equations (4.1.2-1), with the required direction cosines calculated from the quaternion elements as in Equation (3.2.4.2-1).

It should be noted that the heading Euler angle ψ in Equations (4.1.2-1) represents the heading of the B Frame X axis relative to the locally level Frame L. Frame L is sometimes denoted as "platform coordinates" analogous to sensor coordinates on the gyro stabilized element of a gimbaled locally level navigation "platform". Heading is also typically required relative to true North for output purposes (true North is defined as the horizontal direction toward the earth's positive polar rotation axis). To calculate the heading relative to true North (known as "true heading"), ψ must be corrected for the angle between the L Frame X axis and true North (known as the wander angle). Because the wander angle is measured as a positive angle around an upward vertical, but ψ is measured positive around the L Frame downward vertical Z axis, the correction equation has the form:

$$\psi_{True} = \psi_{Platform} - \alpha \qquad \text{(4.1.2-2)}$$

where

ψ_{True} = True heading.

$\psi_{Platform}$ = Platform heading or "ψ" as calculated from Equations (4.1.2-1).

α = Wander angle.

The wander angle is computed from the orientation of the N Frame (or, equivalently, the L Frame using (4.1.1-2)) relative to earth fixed E Frame coordinates, as defined by Equations (4.4.2.1-3).

4.2 ACCELERATION TRANSFORMATION

Before the acceleration measurements from the system accelerometers can be integrated into velocity, the acceleration vector must be transformed from sensor B Frame coordinates to navigation N Frame coordinates. The transformation of measured linear acceleration from the B to the N Frame can be accomplished directly using direction cosine matrix or quaternion attitude parameters by application of Equations (3.2.1-2) or (3.2.4.1-4). Defining the transformation as a two step process (from Frame B to L, then from Frame L to N) we have:

$$\underline{a}_{SF}^{L} = C_B^L \, \underline{a}_{SF}^{B} \tag{4.2-1}$$

or

$$a_{SF}^{L} = q_B^L \, a_{SF}^{B} \, q_B^{L^*} \tag{4.2-2}$$

and then:

$$\underline{a}_{SF}^{N} = C_L^N \, \underline{a}_{SF}^{L} \tag{4.2-3}$$

with C_L^N as the transpose of C_N^L defined by Equation (4.1.1-2), and from Equations (3.2.4.1-3):

$$a_{SF}^{B} = \begin{bmatrix} 0 \\ \underline{a}_{SF}^{B} \end{bmatrix} \qquad a_{SF}^{L} = \begin{bmatrix} 0 \\ \underline{a}_{SF}^{L} \end{bmatrix} \tag{4.2-4}$$

where

$\underline{a}_{SF}^{B}$, $\underline{a}_{SF}^{L}$ = Specific force acceleration vector in the B and L Frames. The linear acceleration measured by accelerometers in the B Frame is $\underline{a}_{SF}^{B}$. The term "specific force" has been adopted for accelerometer measurements to indicate that accelerometers measure acceleration generated by applied contact forces, not including applied gravitational effects. As we shall see, the total acceleration (that produces velocity change) is calculated in the system computer as the sum of specific force and gravitational acceleration.

If a quaternion is being utilized as the basic attitude reference (calculated through integration), Equation (4.2-1) can be used as an alternative to (4.2-2) if the attitude quaternion is first converted to its equivalent direction cosine matrix form as in Equation (3.2.4.2-1):

$$C_B^L = \begin{bmatrix} \left(a^2 + b^2 - c^2 - d^2\right) & 2\,(b\,c - a\,d) & 2\,(b\,d + a\,c) \\ 2\,(b\,c + a\,d) & \left(a^2 - b^2 + c^2 - d^2\right) & 2\,(c\,d - a\,b) \\ 2\,(b\,d - a\,c) & 2\,(c\,d + a\,b) & \left(a^2 - b^2 - c^2 + d^2\right) \end{bmatrix} \tag{4.2-5}$$

The number of individual mathematical operations involved in executing Equations (4.2-5) and (4.2-1) is slightly more than in executing Equation (4.2-2). However, if one is required to also generate Euler angle outputs as in Equation (4.1.2-1), then (4.2-1) with (4.2-5) has less net computations compared with (4.2-2) because many of the C_B^L elements must still be calculated using (4.2-5) for the Equations (4.1.2-1) calculation.

4.3 VELOCITY RATE EQUATION

The velocity data in an inertial navigation system is typically computed as an integration of velocity rate described in the navigation N Frame. The velocity of interest is usually defined as the time rate of change of position relative to earth fixed coordinates (i.e., the E Frame). In Frame E then, we define the velocity vector as:

$$\underline{v}^E \equiv \dot{\underline{R}}^E \tag{4.3-1}$$

where

$\underline{v}^E$ = Column vector representing the defined velocity vector of interest projected along earth E Frame axes.

$\underline{R}^E$ = Column vector representing the position vector from earth's center to the navigation system as viewed in the E Frame.

The components of $\underline{v}$ in locally level N Frame navigation coordinates are the values typically used by the inertial navigation system computer for calculating navigation outputs. These are related to the E Frame components through:

$$\underline{v}^N = C_E^N \; \underline{v}^E \tag{4.3-2}$$

The derivative of (4.3-2) is:

$$\dot{\underline{v}}^N = \dot{C}_E^N \; \underline{v}^E + C_E^N \; \dot{\underline{v}}^E \tag{4.3-3}$$

The $\dot{C}_E^N$ term in (4.3-3) is obtained from the transpose of generalized Equation (3.3.2-6) with (3.2.1-3):

$$\dot{C}_E^N = \left(\underline{\omega}_{EN}^N \times\right)^T C_E^N = -\left(\underline{\omega}_{EN}^N \times\right) C_E^N \tag{4.3-4}$$

in which it is recognized that the transpose of a cross-product operator equals its negative.

The $\dot{\underline{v}}^E$ term in (4.3-3) can be developed by first writing (4.3-1) as a function of $\underline{R}$ components in a non-rotating inertial coordinate I Frame (defined in Section 2.2). Applying (3.4-4) for $\dot{\underline{R}}^E$, we see that (4.3-1) is equivalently:

$$\underline{v}^E = C_I^E \left[\dot{\underline{R}}^I - \left(\underline{\omega}_{IE}^I \times\right) \underline{R}^I\right] \tag{4.3-5}$$

in which it is recognized that $\underline{\omega}_{IE}^I$ is the negative of $\underline{\omega}_{EI}^I$. The $\dot{\underline{v}}^E$ term for Equation (4.3-3) is obtained from the derivative of (4.3-5):

$$\begin{aligned} \dot{\underline{v}}^E &= C_I^E\left[\ddot{\underline{R}}^I - \left(\dot{\underline{\omega}}_{IE}^I \times\right)\underline{R}^I - \left(\underline{\omega}_{IE}^I \times\right)\dot{\underline{R}}^I\right] + \dot{C}_I^E\left[\dot{\underline{R}}^I - \left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I\right] \\ &= C_I^E\left[\ddot{\underline{R}}^I - \left(\underline{\omega}_{IE}^I \times\right)\dot{\underline{R}}^I\right] + \dot{C}_I^E\left[\dot{\underline{R}}^I - \left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I\right] \end{aligned} \tag{4.3-6}$$

The $\dot{\underline{\omega}}_{IE}^I$ term in the previous expression has been equated to zero due to the constancy of earth's rotation rate relative to inertial space. We also know from (3.3.2-6) that:

$$\dot{C}_I^E = C_I^E \left(\underline{\omega}_{EI}^I \times\right) = -C_I^E \left(\underline{\omega}_{IE}^I \times\right) \tag{4.3-7}$$

With (4.3-7), Equation (4.3-6) becomes:

$$\begin{aligned} \dot{\underline{v}}^E &= C_I^E\left[\ddot{\underline{R}}^I - \left(\underline{\omega}_{IE}^I \times\right)\dot{\underline{R}}^I - \left(\underline{\omega}_{IE}^I \times\right)\dot{\underline{R}}^I + \left(\underline{\omega}_{IE}^I \times\right)\left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I\right] \\ &= C_I^E\left[\ddot{\underline{R}}^I - 2\left(\underline{\omega}_{IE}^I \times\right)\dot{\underline{R}}^I + \left(\underline{\omega}_{IE}^I \times\right)\left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I\right] \end{aligned} \tag{4.3-8}$$

The $\dot{\underline{R}}^I$ term in (4.3-8) can be related to $\underline{v}$ through (4.3-5). Multiplication of (4.3-5) by C_E^I and rearrangement yields:

$$\dot{\underline{R}}^I = \underline{v}^I + \left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I \tag{4.3-9}$$

Substitution of (4.3-9) into (4.3-8) obtains:

$$\begin{aligned} \dot{\underline{v}}^E &= C_I^E\left\{\ddot{\underline{R}}^I - 2\left(\underline{\omega}_{IE}^I \times\right)\left[\underline{v}^I + \left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I\right] + \left(\underline{\omega}_{IE}^I \times\right)\left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I\right\} \\ &= C_I^E\left[\ddot{\underline{R}}^I - \left(\underline{\omega}_{IE}^I \times\right)\left(\underline{\omega}_{IE}^I \times\right)\underline{R}^I - 2\left(\underline{\omega}_{IE}^I \times\right)\underline{v}^I\right] \end{aligned} \tag{4.3-10}$$

The $\ddot{\underline{R}}^I$ term in Equation (4.3-10) is the total inertial acceleration which, as discussed in Section 4.2, can be equated to the sum of specific force ($\underline{a}_{SF}$) and gravitational acceleration, the former representing the acceleration sensed by accelerometers:

$$\ddot{\underline{R}}^I = \underline{g}^I + \underline{a}_{SF}^I \tag{4.3-11}$$

where

$\underline{g}^I$ = Gravitational acceleration in I Frame coordinates.

Equation (4.3-10) then becomes:

$$\dot{\underline{v}}^E = C_I^E \left[\underline{a}_{SF}^I + \underline{g}^I - \left(\underline{\omega}_{IE}^I \times \right) \left(\underline{\omega}_{IE}^I \times \right) \underline{R}^I - 2 \left(\underline{\omega}_{IE}^I \times \right) \underline{v}^I \right] \tag{4.3-12}$$

Equation (4.3-12) can be simplified if we analyze its response under zero velocity (and zero velocity rate) conditions which finds:

$$\underline{a}_{SF}^I + \underline{g}^I - \left(\underline{\omega}_{IE}^I \times \right) \left(\underline{\omega}_{IE}^I \times \right) \underline{R}^I = 0 \qquad \text{For } \underline{v} = 0 \tag{4.3-13}$$

or, with (3.1.1-13), for the cross-product vector notation:

$$\underline{a}_{SF}^I = - \left[\underline{g}^I - \underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \underline{R}^I \right) \right] \qquad \text{For } \underline{v} = 0 \tag{4.3-14}$$

from which we define:

$$\underline{g}_P^I \equiv \underline{g}^I - \underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \underline{R}^I \right) \tag{4.3-15}$$

From (4.3-14) and (4.3-13) we see that the $\underline{g}_P^I$ vector is the negative of the specific force acceleration when at rest relative to the earth at position vector $\underline{R}$ from earth's center. We also know that a plumb-bob erected at rest will align itself along the net local force vector on its support pivot. Thus, the direction of $\underline{g}_P^I$ is parallel to the line of a plumb-bob at location $\underline{R}$ in a vehicle that is stationary relative to the earth. For this reason, $\underline{g}_P^I$ is sometimes referred to as "plumb-bob gravity". With this definition, Equation (4.3-12) assumes the simpler form:

$$\dot{\underline{v}}^E = C_I^E \left(\underline{a}_{SF}^I + \underline{g}_P^I - 2\, \underline{\omega}_{IE}^I \times \underline{v}^I \right) \tag{4.3-16}$$

where

$\underline{g}_P^I$ = Plumb-bob gravity in the I Frame as defined by Equation (4.3-15).

We now substitute (4.3-16) and (4.3-4) into (4.3-3) to obtain for $\dot{\underline{v}}^N$:

$$\begin{aligned} \dot{\underline{v}}^N &= -\left(\underline{\omega}_{EN}^N \times\right) C_E^N \, \underline{v}^E + C_E^N \, C_I^E \left[\underline{a}_{SF}^I + \underline{g}_P^I - 2\left(\underline{\omega}_{IE}^I \times\right) \underline{v}^I\right] \\ &= -\left(\underline{\omega}_{EN}^N \times\right) \underline{v}^N + C_I^N \left[\underline{a}_{SF}^I + \underline{g}_P^I - 2\left(\underline{\omega}_{IE}^I \times\right) \underline{v}^I\right] \\ &= -\left(\underline{\omega}_{EN}^N \times\right) \underline{v}^N + \underline{a}_{SF}^N + \underline{g}_P^N - 2\left(\underline{\omega}_{IE}^N \times\right) \underline{v}^N \end{aligned} \qquad (4.3\text{-}17)$$

Introducing (3.1.1-13) for the cross-product vector notation in (4.3-17) and combining terms yields the final expression for $\dot{\underline{v}}^N$:

$$\dot{\underline{v}}^N = \underline{a}_{SF}^N + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N \qquad (4.3\text{-}18)$$

The velocity vector is calculated in the INS computer by integration of Equation (4.3-18) in the N Frame, using $\underline{a}_{SF}^N$ from (4.2-1) - (4.2-3), $\underline{\omega}_{EN}^N$ and $\underline{\omega}_{IE}^N$ from Equations (4.1.1-3) and (4.1.1-6), with $\underline{g}_P^N$ from (4.3-15) calculated as will be shown subsequently in Section 5.4.1.

4.3.1 VELOCITY OUTPUTS

The orientation of the N Frame about the local vertical is somewhat arbitrary (as discussed in Section 4.5), hence, velocity vector components in Frame N have no clear meaning outside the inertial navigation system. For this reason, the velocity components are usually transformed to another universally defined coordinate frame for output purposes.

A typical velocity vector output might be along locally level East (X), North (Y), Upward Vertical (Z) coordinates (denoted here as the Geo Frame). As described in Section 4.4.2.1, the N Frame is rotated from the Geo Frame by the wander angle about the upward vertical. Application of Equations (3.2.3-4) shows that the associated direction cosine transformation matrix is given by:

$$C_N^{Geo} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (4.3.1\text{-}1)$$

where

α = Wander angle.

The velocity vector in the Geo Frame is calculated from:

$$\underline{v}^{Geo} = C_N^{Geo}\,\underline{v}^N \tag{4.3.1-2}$$

Defining:

$$\underline{v}^N = \begin{bmatrix} v_{XN} \\ v_{YN} \\ v_{ZN} \end{bmatrix} \qquad \underline{v}^{Geo} = \begin{bmatrix} v_{East} \\ v_{North} \\ v_{Up} \end{bmatrix} \tag{4.3.1-3}$$

we then obtain after substituting (4.3.1-1) and (4.3.1-3) in (4.3.1-2):

$$\begin{aligned} v_{East} &= v_{XN}\,\cos\alpha - v_{YN}\,\sin\alpha \\ v_{North} &= v_{XN}\,\sin\alpha + v_{YN}\,\cos\alpha \\ v_{Up} &= v_{ZN} \end{aligned} \tag{4.3.1-4}$$

It is important to note that Equations (4.3.1-4) are singular near the polar locations due to wander angle singularities (as discussed in Section 4.4.2.1).

To avoid the Equation (4.3.1-4) singularities, an alternative approach is to calculate velocity components for output in the universally defined earth E Frame using:

$$\underline{v}^E = C_N^E\,\underline{v}^N \tag{4.3.1-5}$$

Defining:

$$\underline{v}^E = \left(v_{XE},\, v_{YE},\, v_{ZE}\right)^T \tag{4.3.1-6}$$

and substituting the C_N^E component and $\underline{v}^N$ definitions from (4.4.1.1-2) into (4.3.1-6) yields the component form:

$$\begin{aligned} v_{XE} &= D_{11}\,v_{XN} + D_{12}\,v_{YN} + D_{13}\,v_{ZN} \\ v_{YE} &= D_{21}\,v_{XN} + D_{22}\,v_{YN} + D_{23}\,v_{ZN} \\ v_{ZE} &= D_{31}\,v_{XN} + D_{32}\,v_{YN} + D_{33}\,v_{ZN} \end{aligned} \tag{4.3.1-7}$$

Note that because the components of C_N^E are cosines, their magnitude will never exceed unity. Hence, Equation (4.3.1-7) has no singularities, and can be used to compute the E Frame velocity vector components for all earth locations.

4.4 POSITION DETERMINATION

Position location can be represented as a linear distance vector from a specified reference point location. For a terrestrial (earth) based inertial navigation system, position is more commonly represented as altitude (h) above the earth's surface coupled with angular position over the earth's surface. Angular position over the earth can be described by the third column of the navigation Frame N to earth Frame E direction cosine matrix (C_N^E) which represents the projection of a locally vertical unit vector on earth E Frame axes. Alternatively, earth angular position can be represented by latitude and longitude Euler angles. In general, position location in any desired analytical form can be computed from another previously determined position representation. In a typical inertial navigation system, h is calculated by integrating vertical velocity, and C_N^E is obtained from an integration operation using $\underline{\omega}_{EN}^N$ as input. Other position parameters are then calculated from the computed C_N^E and h.

In this section we will derive the equations for calculating C_N^E and h by integration, and then develop analytical expressions relating the various position representation forms. The altitude integration discussion will include methods for dealing with altitude computational instabilities generated by vertical velocity error (excited principally from accelerometer input error). The subsection on latitude/longitude extraction from C_N^E includes calculation of the wander angle relating N Frame and local geographic Geo Frame horizontal axes. As an exercise, this section also addresses the alternative of calculating latitude and longitude by integrating latitude/longitude rate equations. As will be shown, this generally is not preferred due to singularities at the earth polar regions.

4.4.1 <u>TYPICAL POSITION RATE EQUATIONS</u>

Differential rate equations for the position direction cosine matrix C_N^E and altitude h are derived in the following subsections.

4.4.1.1 POSITION DIRECTION COSINE MATRIX (FRAME N TO E) RATE EQUATIONS

The C_N^E matrix changes as a function of the angular rotation of Frame N relative to Frame E as prescribed by the angular transport rate vector $\underline{\rho}^N$ calculated in Equation (4.1.1-6). The associated differential equation for C_N^E is obtained by general application of Equation (3.3.2-6) using $\underline{\rho}^N$ for $\underline{\omega}_{EN}^N$ as in (4.1.1-6):

$$\dot{C}_N^E = C_N^E \left(\underline{\rho}^N \times\right) \tag{4.4.1.1-1}$$

The component form of Equation (4.4.1.1-1) for integration is obtained by substituting the following definitions for the matrix and vector components:

$$\underline{\rho}^N \equiv \underline{\omega}_{EN}^N = \begin{bmatrix} \rho_{XN} \\ \rho_{YN} \\ \rho_{ZN} \end{bmatrix} \qquad C_N^E = \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix} \tag{4.4.1.1-2}$$

The result is:

$$\begin{aligned}
\dot{D}_{11} &= D_{12}\,\rho_{ZN} - D_{13}\,\rho_{YN} \\
\dot{D}_{12} &= D_{13}\,\rho_{XN} - D_{11}\,\rho_{ZN} \\
\dot{D}_{13} &= D_{11}\,\rho_{YN} - D_{12}\,\rho_{XN} \\
\\
\dot{D}_{21} &= D_{22}\,\rho_{ZN} - D_{23}\,\rho_{YN} \\
\dot{D}_{22} &= D_{23}\,\rho_{XN} - D_{21}\,\rho_{ZN} \\
\dot{D}_{23} &= D_{21}\,\rho_{YN} - D_{22}\,\rho_{XN} \\
\\
\dot{D}_{31} &= D_{32}\,\rho_{ZN} - D_{33}\,\rho_{YN} \\
\dot{D}_{32} &= D_{33}\,\rho_{XN} - D_{31}\,\rho_{ZN} \\
\dot{D}_{33} &= D_{31}\,\rho_{YN} - D_{32}\,\rho_{XN}
\end{aligned} \tag{4.4.1.1-3}$$

As with the calculation of the C_B^L components discussed in Section 4.1, the third row of the C_N^E matrix can alternatively be computed from the first two rows using the principle that each row of C_N^E represents a unit vector along Frame E axes, hence, the cross product of rows one and two equals row three. Therefore:

$$\begin{aligned}
D_{31} &= D_{12}\,D_{23} - D_{13}\,D_{22} \\
D_{32} &= D_{13}\,D_{21} - D_{11}\,D_{23} \\
D_{33} &= D_{11}\,D_{22} - D_{12}\,D_{21}
\end{aligned} \tag{4.4.1.1-4}$$

A unit vector along the local vertical (i.e., the Z axis of the N Frame) projected on E Frame axes (which from Equations (3.2.1-6) equals the third column of C_N^E) uniquely defines the angular earth surface position location. Equation (3.3.1-2) applied to the third column of C_N^E (using $\underline{\rho}$ for $\underline{\omega}_{EN}$) is:

$$\dot{\underline{u}}_{ZN}^E = \underline{\rho}^E \times \underline{u}_{ZN}^E \tag{4.4.1.1-5}$$

where

$\underline{u}_{ZN}^E$ = Unit vector along the N Frame Z axis projected on the E Frame (i.e., the third column of C_N^E).

We can also express $\underline{\rho}^E$ as the sum of its vertical and horizontal components:

$$\underline{\rho}^E = \underline{\rho}_H^E + \rho_{ZN}\, \underline{u}_{ZN}^E \tag{4.4.1.1-6}$$

where

H = Designation for horizontal components.

Applying (4.4.1.1-6) to (4.4.1.1-5) shows that:

$$\dot{\underline{u}}_{ZN}^E = \underline{\rho}_H^E \times \underline{u}_{ZN}^E \tag{4.4.1.1-7}$$

From (4.4.1.1-7) we see that the vertical $\underline{\rho}^N$ component (ρ_{ZN}) has no effect on the position determination function and can be arbitrarily selected to enhance overall navigation equation characteristics.

One of the principal characteristics desired for ρ_{ZN} and for all components of $\underline{\rho}^N$, is that they have no singularities for any local level Frame N location. Since the D_{IJ} elements in Equations (4.4.1.1-3) can never exceed unity in magnitude (because they are direction cosines), non-singular behavior for the components of $\underline{\rho}^N$ assures that Equations (4.4.1.1-3) can be integrated free of singularities for any Frame N attitude relative to earth Frame E. Section 4.5 discusses the selection of ρ_{ZN} to eliminate singularities. The horizontal components of $\underline{\rho}^N$, represented by the $F_C^N \left(\underline{u}_{ZN}^N \times \underline{v}^N\right)$ term in (4.1.1-6), are inherently free of singularities for finite velocities.

4.4.1.2 ALTITUDE RATE EQUATION

The equation for altitude rate is obtained from the derivative of the defining equation for altitude:

$$\underline{h}^N = h\,\underline{u}_{ZN}^N = \underline{R}^N - \underline{R}_S^N = \underline{u}_{ZN}^N \cdot \left(\underline{R}^N - \underline{R}_S^N\right) \tag{4.4.1.2-1}$$

where

h = Altitude.

$\underline{u}_{ZN}^N$ = Unit vector (in N Frame axes) that is perpendicular (along the upward local vertical) to the earth surface and is directed through the INS position point. By the definition of the N Frame, $\underline{u}_{ZN}^N$ is along the N Frame vertical axis Z.

$\underline{h}^N$ = Altitude vector.

$\underline{R}^N$ = Position vector from earth center to the INS.

$\underline{R}_S^N$ = Position vector from earth center to the earth surface point from which $\underline{u}_{ZN}^N$ emanates.

The altitude rate is the derivative of h in (4.4.1.2-1):

$$\dot{h} = \underline{u}_{ZN}^N \cdot \left(\dot{\underline{R}}^N - \dot{\underline{R}}_S^N\right) \tag{4.4.1.2-2}$$

in which it is recognized that the rate of change of $\underline{u}_{ZN}^N$ is zero because it is defined as a unit vector along the N Frame vertical axis (hence, its derivative in the N Frame is zero).

We can also write:

$$\underline{R}^N = C_E^N\,\underline{R}^E \qquad \underline{R}_S^N = C_E^N\,\underline{R}_S^E \tag{4.4.1.2-3}$$

The derivative of (4.4.1.2-3) is:

$$\dot{\underline{R}}^N = C_E^N\,\dot{\underline{R}}^E + \dot{C}_E^N\,\underline{R}^E \qquad \dot{\underline{R}}_S^N = C_E^N\,\dot{\underline{R}}_S^E + \dot{C}_E^N\,\underline{R}_S^E \tag{4.4.1.2-4}$$

The $\dot{C}_E^N$ term in (4.4.1.2-4) is given by (4.3-4). Equations (4.3-4) and (4.3-1) inserted in (4.4.1.2-4) obtains:

$$\dot{\underline{R}}^N = C_E^N\,\underline{v}^E - \left(\underline{\omega}_{EN}^N \times\right) C_E^N\,\underline{R}^E \qquad \dot{\underline{R}}_S^N = C_E^N\,\dot{\underline{R}}_S^E - \left(\underline{\omega}_{EN}^N \times\right) C_E^N\,\underline{R}_S^E \tag{4.4.1.2-5}$$

or

$$\dot{\underline{R}}^N = \underline{v}^N - \underline{\omega}_{EN}^N \times \underline{R}^N \qquad\qquad \dot{\underline{R}}_S^N = C_E^N \, \dot{\underline{R}}_S^E - \underline{\omega}_{EN}^N \times \underline{R}_S^N \tag{4.4.1.2-6}$$

Substituting (4.4.1.2-6) with (4.4.1.2-1) into (4.4.1.2-2) yields:

$$\begin{aligned} \dot{h} &= \underline{u}_{ZN}^N \cdot \left[\underline{v}^N - C_E^N \, \dot{\underline{R}}_S^E - \underline{\omega}_{EN}^N \times \left(\underline{R}^N - \underline{R}_S^N \right) \right] \\ &= \underline{u}_{ZN}^N \cdot \left[\underline{v}^N - C_E^N \, \dot{\underline{R}}_S^E - h \left(\underline{\omega}_{EN}^N \times \underline{u}_{ZN}^N \right) \right] \\ &= \underline{u}_{ZN}^N \cdot \underline{v}^N - \underline{u}_{ZN}^N \cdot \left(C_E^N \, \dot{\underline{R}}_S^E \right) \end{aligned} \tag{4.4.1.2-7}$$

From the definition of $\underline{R}_S$ as a vector from earth's center to the local earth surface, changes in $\underline{R}_S^E$ produced by vehicle translation must be horizontal along the earth surface. As such, the second term in Equation (4.4.1.2-7) is identically zero. This is also demonstrated rigorously in Section 5.5 of the next chapter. The final equation for altitude rate, therefore, is:

$$\dot{h} = \underline{u}_{ZN}^N \cdot \underline{v}^N \tag{4.4.1.2-8}$$

4.4.1.2.1 Vertical Channel Control

For applications with navigation times exceeding 10 minutes, the vertical position error generated within a pure inertial navigation system (due primarily to the integrated effect of accelerometer errors) typically exceeds system accuracy requirements. For operation at positive altitudes, the inertially computed vertical position error is amplified for extended navigation times due to a divergent error characteristic that grows exponentially with time. The divergence effect stems from the $\underline{g}_P$ plumb-bob gravity term in (4.3-18) which is the sum of mass attraction and earth rate centripetal accelerations (see Equation (4.3-15)). The mass attraction term is the dominant effect and is a calculated function of position in the navigation computer. For positive altitudes, the magnitude of mass attraction gravity varies inversely as the square of distance from earth's center (see Section 5.4 - Equation (5.4-1)). Consequently, altitude errors generated in the inertial altitude computation process (i.e., the integration of Equations (4.3-18) and (4.4.1.2-8)) will directly impact the vertical gravity component magnitude in (4.3-18). Positive altitude error for example, will produce a smaller than actual gravity magnitude (i.e., a negative gravity magnitude error), which decreases total acceleration downward, hence, increases vertical acceleration upward. Increased upward vertical acceleration generates increased positive upward velocity which in turn generates larger positive altitude error. The effect increases exponentially with time.

If error analysis finds the vertical position error unacceptable (invariably the case for most applications with navigation times exceeding 10 minutes), external inputs must be provided to eliminate the error in vertical velocity and altitude. One of the classic methods for eliminating vertical channel error in aircraft inertial navigation systems is through application of the barometric pressure altitude signal provided as input to the INS from the air data computer (for an underwater vehicle INS application, the equivalent input would be provided by the pressure depth transducer signal). The method is to compare the inertial and pressure altitudes, with the difference used through suitable gains in servo fashion as feedback to the vertical velocity/altitude integrators. A typical implementation of the approach is defined analytically as an extension of Equations (4.3-18) and (4.4.1.2-8):

$$\dot{\underline{v}}^N = \underline{a}_{SF}^N + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N - e_{vc_1}\,\underline{u}_{ZN}^N \tag{4.4.1.2.1-1}$$

$$\dot{h} = \underline{v}^N \cdot \underline{u}_{ZN}^N - e_{vc_2} \tag{4.4.1.2.1-2}$$

$$\begin{aligned} &\partial h = h - h_{Prsr} \\ &e_{vc_1} = e_{vc_3} + C_2\,\partial h \qquad e_{vc_2} = C_3\,\partial h \qquad \dot{e}_{vc_3} = C_1\,\partial h \end{aligned} \tag{4.4.1.2.1-3}$$

where

$e_{vc_1}, e_{vc_2}, e_{vc_3}$ = Vertical channel control signals.

C_1, C_2, C_3 = Vertical channel control gains.

h_{Prsr} = Pressure altitude input signal.

∂h = Altitude error signal.

The e_{vc_3} signal (the integral of $\dot{e}_{vc_3}$ in Equations (4.4.1.2.1-3)), is incorporated as an integral controller to generate a bias offset for vertical accelerometer error present on the $\underline{a}_{SF}^N$ signal feeding $\dot{\underline{v}}^N$. Without e_{vc_3}, the servo characteristics of Equations (4.4.1.2.1-1) - (4.4.1.2.1-3) would develop an offset in ∂h to balance the vertical accelerometer error (through the C_2 gain in e_{vc_1}). This can be seen by analyzing the error form (differential) of the vertical component of Equations (4.4.1.2.1-1) - (4.4.1.2.1-3):

$$\begin{aligned} &\delta\dot{v}_{ZN} = \delta a_{SF_{ZN}} - \delta e_{vc_1} + \cdots \qquad && \delta\dot{h} = \delta v_{ZN} - \delta e_{vc_2} \\ &\delta e_{vc_1} = \delta e_{vc_3} + C_2\left(\delta h - \delta h_{Prsr}\right) && \delta e_{vc_2} = C_3\left(\delta h - \delta h_{Prsr}\right) \\ &\qquad \delta\dot{e}_{vc_3} = C_1\left(\delta h - \delta h_{Prsr}\right) \end{aligned} \tag{4.4.1.2.1-4}$$

where

$\delta(\;)$ = Error in the indicated parameter.

$\delta h, \delta h_{Prsr}, \delta v_{ZN}, \delta a_{SF_{ZN}}$ = Inertial altitude error, pressure altitude error, the vertical component of the velocity error vector (along the Z axis of the N Frame), and the vertical (ZN) component of the specific force acceleration error vector (e.g., from accelerometers).

$\cdots$ = Terms that are generally small compared to the other terms, hence, negligible for the discussion at hand.

We then combine Equations (4.4.1.2.1-4) for the case when the integral controller is absent (i.e., $C_1 = 0$). Differentiating $\delta\dot{h}$ and substituting $\delta\dot{v}_{ZN}$ in the result finds (for zero C_1):

$$\delta\ddot{h} = \delta a_{SF_{ZN}} - \delta e_{vc_1} - \delta\dot{e}_{vc_2} + \cdots = \delta a_{SF_{ZN}} - C_2 \left(\delta h - \delta h_{Prsr}\right) - C_3 \frac{d}{dt}\left(\delta h - \delta h_{Prsr}\right) + \cdots$$

or

$$\frac{d^2}{dt^2}\left(\delta h - \delta h_{Prsr}\right) + C_3 \frac{d}{dt}\left(\delta h - \delta h_{Prsr}\right) + C_2 \left(\delta h - \delta h_{Prsr}\right) = \delta a_{SF_{ZN}} - \frac{d^2}{dt^2}\delta h_{Prsr} + \cdots \tag{4.4.1.2.1-5}$$

where

$\ddot{(\;)}$ = Second time derivative of parameter in brackets.

Equation (4.4.1.2.1-5) shows that for stable C_2 and C_3 gains (and slowly changing δh_{Prsr}), in the steady state (i.e., when the derivative terms become zero), the $C_2 \left(\delta h - \delta h_{Prsr}\right)$ term balances $\delta a_{SF_{ZN}}$, which forces an offset into δh (in addition to the normal offset in δh equal to δh_{Prsr}). Let's consider the case when C_1 is non-zero. Double differentiating $\delta\dot{h}$ in (4.4.1.2.1-4) and substituting the derivative of $\delta\dot{v}_{ZN}$ finds:

$$\begin{aligned}\dddot{\delta h} &= \delta\dot{a}_{SF_{ZN}} - \delta\dot{e}_{vc_1} - \delta\ddot{e}_{vc_2} + \cdots = \delta\dot{a}_{SF_{ZN}} - \delta\dot{e}_{vc_3} \\ &\quad - C_2 \frac{d}{dt}\left(\delta h - \delta h_{Prsr}\right) - C_3 \frac{d^2}{dt^2}\left(\delta h - \delta h_{Prsr}\right) + \cdots\end{aligned}$$

or

$$\begin{aligned}&\frac{d^3}{dt^3}\left(\delta h - \delta h_{Prsr}\right) + C_3 \frac{d^2}{dt^2}\left(\delta h - \delta h_{Prsr}\right) + C_2 \frac{d}{dt}\left(\delta h - \delta h_{Prsr}\right) \\ &\qquad + C_1 \left(\delta h - \delta h_{Prsr}\right) = \delta\dot{a}_{SF_{ZN}} - \delta\dddot{h}_{Prsr} + \cdots\end{aligned} \tag{4.4.1.2.1-6}$$

where

$\dddot{(\;)}$ = Third time derivative of parameter in brackets.

Now in the steady state (and slowly changing δh_{Prsr}), $C_1 \left(\delta h - \delta h_{Prsr}\right)$ balances $\delta \dot{a}_{SF_{ZN}}$, hence, for steady $\delta a_{SF_{ZN}}$, the δh altitude error goes to δh_{Prsr}, the expected nominal steady state condition. Further analysis also shows that with C_1 active, the $\delta a_{SF_{ZN}}$ error becomes balanced by the δe_{vc_3} integrator (look at the $\delta \dot{v}_{ZN}$ expression in (4.4.1.2.1-4) which, when zero in the steady state, sets δe_{vc_1} to $\delta a_{SF_{ZN}}$. Substituting for δe_{vc_1} from (4.4.1.2.1-4) with $\delta h = \delta h_{Prsr}$ in the steady state shows that $\delta e_{vc_3} = \delta a_{SF_{ZN}}$).

An important characteristic inherent in the structure of Equations (4.4.1.2.1-1) - (4.4.1.2.1-3) is that under error free conditions (i.e., $\partial h = 0$), they revert back to the classical forms of Equations (4.3-18) and (4.4.1.2-8). Hence, the feedback terms in (4.4.1.2.1-1) and (4.4.1.2.1-2) only operate on the vertical velocity/altitude errors. The true (correct) velocity/altitude signals are unaffected by the feedback. In some implementations, the inertial altitude h (used in calculating ∂h) is filtered to match the dynamic characteristics of the h_{Prsr} input signal, thereby enhancing the nullification of ∂h feedback under zero error conditions.

Proper dynamic characteristics for Equations (4.4.1.2.1-1) - (4.4.1.2.1-3) are typically designed using classical servo control theory for root placement. For example, the characteristic roots of differential Equation (4.4.1.2.1-6) are classically obtained (as in Reference 38, Sections 18-8 through 18-11) by evaluating the homogeneous portion; i.e., setting the left side equal to zero with:

$$\delta h - \delta h_{Prsr} = A\, e^{\lambda t} \tag{4.4.1.2.1-7}$$

where

A = Constant.

λ = Characteristic root of the differential equation.

The result is:

$$\left(\lambda^3 + C_3 \lambda^2 + C_2 \lambda + C_1\right) A = 0 \tag{4.4.1.2.1-8}$$

or

$$\lambda^3 + C_3 \lambda^2 + C_2 \lambda + C_1 = 0 \tag{4.4.1.2.1-9}$$

Equation (4.4.1.2.1-9) is the characteristic equation for (4.4.1.2.1-6) that characterizes the dynamic response. If we desire that the dynamic response be characterized by one real root and one resonant root pair we can write:

$$\begin{aligned} \lambda^3 + C_3 \lambda^2 + C_2 \lambda + C_1 &= \left(\lambda + \frac{1}{\tau}\right)\left(\lambda^2 + 2\,\zeta\,\omega_n\,\lambda + \omega_n^2\right) \\ &= \lambda^3 + \left(\frac{1}{\tau} + 2\,\zeta\,\omega_n\right)\lambda^2 + \left(\frac{2\,\zeta\,\omega_n}{\tau} + \omega_n^2\right)\lambda + \frac{\omega_n^2}{\tau} \end{aligned} \tag{4.4.1.2.1-10}$$

where

τ = Time constant for the real root.

$\zeta,\ \omega_n$ = Damping ratio and undamped natural frequency for the resonant root pair.

Equating coefficients of like powers of λ in Equation (4.4.1.2.1-10) then yields:

$$C_1 = \frac{\omega_n^2}{\tau} \qquad C_2 = \frac{2\,\zeta\,\omega_n}{\tau} + \omega_n^2 \qquad C_3 = \frac{1}{\tau} + 2\,\zeta\,\omega_n \tag{4.4.1.2.1-11}$$

The magnitude of the C_1, C_2, C_3 control gains (determined in (4.4.1.2.1-11) by the selected ζ, ω_n and τ) are set large enough to adequately attenuate expected accelerometer errors, but small enough to avoid introducing high frequency h_{Prsr} error effects into the inertial vertical velocity/altitude (through ∂h). This tradeoff can be quantified from the Laplace transform solution to Equations (4.4.1.2.1-4). First we combine (4.4.1.2.1-4) terms and differentiate the $\delta \dot{v}_{ZN}$ expression:

$$\begin{aligned} \delta \ddot{v}_{ZN} &= \delta \dot{a}_{SF_{ZN}} - \delta \dot{e}_{vc_1} + \cdots = \delta \dot{a}_{SF_{ZN}} - \delta \dot{e}_{vc_3} - C_2 \frac{d}{dt}\left(\delta h - \delta h_{Prsr}\right) + \cdots \\ &= \delta \dot{a}_{SF_{ZN}} - C_1 \left(\delta h - \delta h_{Prsr}\right) - C_2 \frac{d}{dt}\left(\delta h - \delta h_{Prsr}\right) + \cdots \\ \delta \dot{h} &= \delta v_{ZN} - \delta e_{vc_2} = \delta v_{ZN} - C_3 \left(\delta h - \delta h_{Prsr}\right) \end{aligned} \tag{4.4.1.2.1-12}$$

Then we take the Laplace transform, neglecting the initial condition terms (as decaying to zero):

$$\begin{aligned} S^2\, \delta V_{ZN}(S) &= \delta \dot{A}_{SF_{ZN}}(S) - \left(C_1 + C_2\, S\right)\left(\delta H(S) - \delta H_{Prsr}(S)\right) + \cdots \\ S\left(\delta H(S) - \delta H_{Prsr}(S)\right) &= \delta V_{ZN}(S) - S\,\delta H_{Prsr}(S) - C_3\left(\delta H(S) - \delta H_{Prsr}(S)\right) \end{aligned} \tag{4.4.1.2.1-13}$$

where

S = Laplace transform parameter.

$\delta H(S)$, $\delta V_{ZN}(S)$, $\delta H_{Prsr}(S)$, $\delta \dot{A}_{SF_{ZN}}(S)$ = Laplace transforms of δh, δv_{ZN}, δh_{Prsr}, and $\delta \dot{a}_{SF_{ZN}}$.

The second expression in (4.4.1.2.1-13) is by rearrangement:

$$\delta H(S) - \delta H_{Prsr}(S) = \frac{1}{S + C_3}\Big(\delta V_{ZN}(S) - S\ \delta H_{Prsr}(S)\Big) \qquad (4.4.1.2.1\text{-}14)$$

Combining the first expression in (4.4.1.2.1-13) with (4.4.1.2.1-14) finds:

$$S^2\ \delta V_{ZN}(S) = \delta\dot{A}_{SF_{ZN}}(S) - \frac{C_1 + C_2\ S}{S + C_3}\Big(\delta V_{ZN}(S) - S\ \delta H_{Prsr}(S)\Big) + \cdots \qquad (4.4.1.2.1\text{-}15)$$

Rearranging:

$$\Big[S^2\,(S + C_3) + (C_1 + C_2\ S)\Big]\ \delta V_{ZN}(S) = (S + C_3)\ \delta\dot{A}_{SF_{ZN}}(S) + (C_1 + C_2\ S)\ S\ \delta H_{Prsr}(S) + \cdots$$

and solving for $\delta V_{ZN}(S)$ gives:

$$\begin{aligned} \delta V_{ZN}(S) = {} & \frac{S\,(C_2\ S + C_1)}{S^3 + C_3\ S^2 + C_2\ S + C_1}\ \delta H_{Prsr}(S) \\ & + \frac{S + C_3}{S^3 + C_3\ S^2 + C_2\ S + C_1}\ \delta\dot{A}_{SF_{ZN}}(S) + \cdots \end{aligned} \qquad (4.4.1.2.1\text{-}16)$$

The solution for $\delta H(S)$ is found most directly from the Laplace transform of (4.4.1.2.1-6):

$$\Big(S^3 + C_3\ S^2 + C_2\ S + C_1\Big)\Big(\delta H(S) - \delta H_{Prsr}(S)\Big) = - S^3\ \delta H_{Prsr}(S) + \delta\dot{A}_{SF_{ZN}}(S) + \cdots$$

Solving for $\delta H(S)$ then yields:

$$\begin{aligned} \delta H(S) = {} & \frac{C_3\ S^2 + C_2\ S + C_1}{S^3 + C_3\ S^2 + C_2\ S + C_1}\ \delta H_{Prsr}(S) \\ & + \frac{1}{S^3 + C_3\ S^2 + C_2\ S + C_1}\ \delta\dot{A}_{SF_{ZN}}(S) + \cdots \end{aligned} \qquad (4.4.1.2.1\text{-}17)$$

Equations (4.4.1.2.1-16) and (4.4.1.2.1-17) describe the vertical velocity and altitude error Laplace transform response to pressure altitude error and vertical accelerometer component error. For zero control gains, these equations collapse into the inertial solution in which the vertical velocity and altitude errors equal the double and triple integral of the vertical acceleration error rate (i.e., the inverse Laplace transform of $\frac{1}{S^2}\ \delta\dot{A}_{SF_{ZN}}(S)$ and $\frac{1}{S^3}\ \delta\dot{A}_{SF_{ZN}}(S)$, respectively). For active control gains and constant vertical acceleration error, $\delta\dot{A}_{SF_{ZN}}(S)$ is zero and there is no vertical channel response to accelerometer error. For very large gains, the $\delta\dot{A}_{SF_{ZN}}(S)$ error contribution to vertical channel response is attenuated, but the $\delta H_{Prsr}(S)$ error couples directly into altitude error, and as the derivative, into vertical velocity error (i.e., the inverse Laplace transform of $S\,\delta H_{Prsr}(S)$). The vertical channel error response to frequency content in $\delta H_{Prsr}(S)$ and $\delta\dot{A}_{SF_{ZN}}(S)$ can be assessed using Section 10.2.1 (Equations (10.2.1-1) - (10.2.1-4), (10.2.1-14), (10.2.1-17) and (10.2.1-26)) which show that the amplitude response of a linear system to sinusoidal inputs equals its Laplace transform with $j\,\omega$ substituted for the Laplace parameter S, where

j = The imaginary parameter that equals the square root of minus one.

ω = Input sinusoid frequency.

Substituting $j\,\omega$ for S in (4.4.1.2.1-16) and (4.4.1.2.1-17), we see that high frequency (ω) components in $\delta\dot{A}_{SF_{ZN}}(S)$ are naturally attenuated by the dominant $\frac{S}{S^3} = \frac{1}{S^2}$ term in (4.4.1.2.1-16) and by $\frac{1}{S^3}$ in (4.4.1.2.1-17) (i.e., independent of control gain selection). The low frequency components in $\delta\dot{A}_{SF_{ZN}}(S)$ are attenuated by $\frac{C_3}{C_1}$ in (4.4.1.2.1-16) and by $\frac{1}{C_1}$ in (4.4.1.2.1-17). For a strapdown inertial navigation system, low frequency errors in $\delta\dot{A}_{SF_{ZN}}(S)$ are caused by maneuvering flight conditions that change the accelerometer error magnitudes (e.g., scale factor errors) and rotate different strapdown accelerometer error components into the vertical. For the $\delta H_{Prsr}(S)$ error, we see that high frequency components (i.e., noise) are attenuated in the (4.4.1.2.1-16) velocity error equation by $\frac{S\,C_2\,S}{S^3} = \frac{C_2}{S}$. Similarly, the (4.4.1.2.1-17) altitude error response to high frequency $\delta H_{Prsr}(S)$ error components is attenuated by $\frac{C_3\,S^3}{S^4} = \frac{C_3}{S}$. For low frequency errors in $\delta H_{Prsr}(S)$, the $\frac{S\,C_1}{C_1} = S$ term in the (4.4.1.2.1-16) velocity error equation couples the derivative of noise directly into the vertical velocity error, and $\frac{C_1}{C_1} = 1$ in (4.4.1.2.1-17) couples the noise directly into the vertical position error (i.e., both independent of control gain selection except for gains approaching zero for which $\delta H_{Prsr}(S)$ coupling goes to zero). Thus, gain selection is based on a tradeoff between higher control gains to attenuate low frequency accelerometer error effects versus lower control gains to minimize high frequency pressure altitude noise coupling into the vertical channel.

4.4.2 POSITION PARAMETER EQUIVALENCIES

The following subsections derive analytical equivalencies between different position location representations. The E and N Frames used in these Sections are as defined in Sections 2.2 and 4.0.

4.4.2.1 LATITUDE/LONGITUDE FROM POSITION (N TO E) DIRECTION COSINE MATRIX

Latitude and longitude can be calculated from the C_N^E direction cosine matrix if we define the angular relationship between the local level N Frame and earth fixed equatorial E Frame as illustrated in Figure 4.4.2.1-1 by the Equation (3.2.3-1) Euler sequence:

$$
\begin{aligned}
\underline{u}_{\phi} &= \underline{u}_{YE} \qquad & \phi &= L \\
\underline{u}_{\phi} &= \underline{u}_{XA_1} \qquad & \phi &= -\,l \\
\underline{u}_{\phi} &= \underline{u}_{ZA_2} \qquad & \phi &= \alpha
\end{aligned}
\qquad (4.4.2.1\text{-}1)
$$

where:

$\underline{u}_{\phi}$ = Generalized Euler angle rotation axis.

ϕ = Generalized Euler rotation angle about $\underline{u}_{\phi}$.

L = Current longitude measured in the equatorial plane, from the intersection of the equatorial plane with the Greenwich Prime Meridian reference plane as a positive rotation about the earth polar axis (i.e., the E Frame Y axis).

l = Current latitude measured positive from the equator in the Northern hemisphere.

α = Wander angle between North and the local level N Frame Y axis, measured as a positive rotation from North about the upward vertical (i.e., about the N Frame Z axis).

A_1 = Coordinate Frame rotated from the earth Frame E by the longitude Euler angle L.

A_2 = Coordinate Frame rotated from the A_1 Frame by minus the latitude Euler angle l.

$\underline{u}_{YE}$, $\underline{u}_{XA_1}$, $\underline{u}_{ZA_2}$ = Unit vectors along the Frame E Y-axis, Frame A_1 X-axis and Frame A_2 Z-axis.

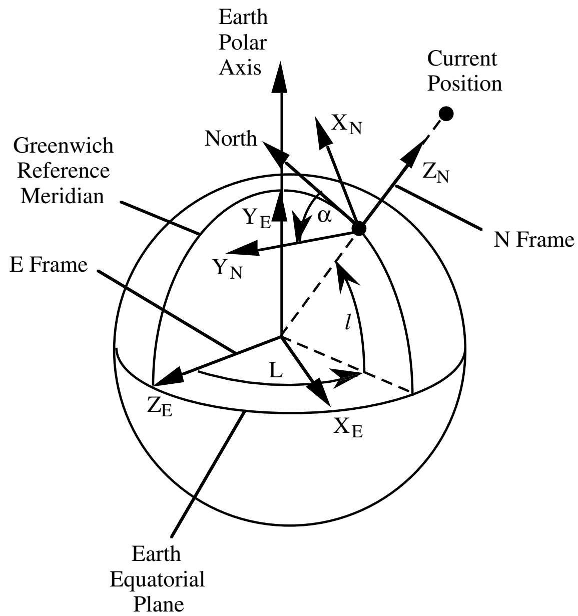


Figure 4.4.2.1-1 Latitude, Longitude, Wander Angle Description

Applying the Section 3.2.3.3 Method of Least Work to the previous Euler sequence yields:

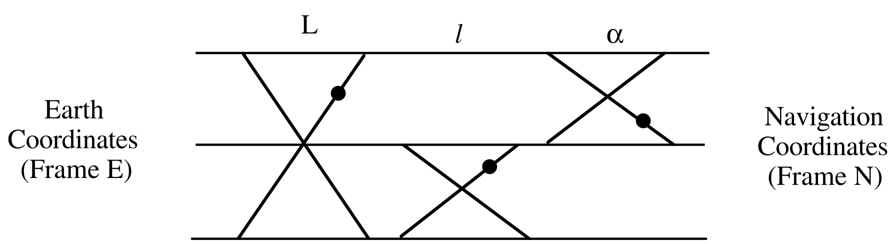


Figure 4.4.2.1-2 Latitude, Longitude, Wander Angle Method Of Least Work Diagram

Note in Figure 4.4.2.1-2 that the dot (·) is inverted from Figure 3.2.3.3 on the l rotation because it is defined about the minus X axis (see explanation following Figure 3.2.3.3-3).

We can now easily obtain a set of equations for the direction cosines between the earth Frame E and the local level navigation Frame N in terms of L, l and α. Using the Section

3.2.3.3 "Method" and the definition in Equations (4.4.1.1-2) for the C_N^E direction cosine elements we obtain:

$$
\begin{aligned}
D_{11} &= \cos L \cos \alpha - \sin L \sin l \sin \alpha \\
D_{12} &= -\cos L \sin \alpha - \sin L \sin l \cos \alpha \\
D_{13} &= \sin L \cos l \\[6pt]
D_{21} &= \cos l \sin \alpha \\
D_{22} &= \cos l \cos \alpha \\
D_{23} &= \sin l \\[6pt]
D_{31} &= -\sin L \cos \alpha - \cos L \sin l \sin \alpha \\
D_{32} &= \sin L \sin \alpha - \cos L \sin l \cos \alpha \\
D_{33} &= \cos L \cos l
\end{aligned}
\qquad (4.4.2.1\text{-}2)
$$

from which the Euler angles are extracted by inversion:

$$
l = \tan^{-1}\left(\frac{D_{23}}{\sqrt{D_{21}^2 + D_{22}^2}}\right) \qquad L = \tan^{-1}\left(\frac{D_{13}}{D_{33}}\right) \qquad \alpha = \tan^{-1}\left(\frac{D_{21}}{D_{22}}\right) \qquad (4.4.2.1\text{-}3)
$$

The plus sign chosen for the square root in the latitude (l) expression in (4.4.2.1-3) corresponds to selecting latitude to be in the range $-\frac{\pi}{2} \leq l \leq \frac{\pi}{2}$ by definition.

The wander angle α in Equations (4.4.2.1-3) is used to transform horizontal vector data in Frame N (such as the horizontal N Frame velocity components in Section 4.3.1) to the equivalent North/East component forms, typically for output purposes. Additionally, heading angle data referenced to the N Frame (such as ψ in Equations (4.1.2-1) known as "platform heading") can be referenced to North by subtracting the wander angle (as in (4.1.2-2)), thereby obtaining what is known as "true heading".

It should be noted that longitude L and wander angle α in Equations (4.4.2.1-3) are singular for latitude l equal to $\pm \pi/2$ (i.e., at the North and South poles). From Equations (4.4.2.1-2), D_{13}, D_{33}, D_{21}, and D_{22} become zero under these conditions making their ratios indeterminate in Equations (4.4.2.1-3). This is directly analogous to the aircraft pitch Euler angle singularity condition discussed in Section 3.2.3.2 caused by the A Frame Z axis and B Frame X axis becoming parallel. In Equations (4.4.2.1-3), the singularity is caused by the E Frame Y axis and N Frame Z axis becoming parallel. Under this condition, longitude and wander angle are indistinguishable from one another and the concept of a North/South direction becomes meaningless (i.e., when at the North pole all directions are South and when at the South pole all directions are North). From a navigational standpoint, the solution is to use another set of parameters to define earth surface position and heading reference when in the polar regions (such as "Grid" coordinates which define position in a horizontal grid coordinate system having grid lines parallel and perpendicular to the Prime Meridian reference plane).

4.4.2.2 POSITION VECTOR IN SELECTED EARTH FIXED FRAME FROM LATITUDE, LONGITUDE, ALTITUDE

Frequently it is desired to express position as the distance from a specified earth fixed reference point location to the current position location, expressed as a vector in a selected earth fixed coordinate frame. For this development, we shall define the selected frame as being parallel to east, north, up directions at the specified earth fixed reference location point. We begin the development by first defining:

$$\Delta\underline{R} = \underline{R} - \underline{R}_{REF} \tag{4.4.2.2-1}$$

where

$\underline{R}$ = Position vector from earth's center to the current position location.

$\underline{R}_{REF}$ = Position vector from earth's center to the specified reference position location.

$\Delta\underline{R}$ = Position vector from the specified reference position location to the current position location.

The $\underline{R}$ and $\underline{R}_{REF}$ vectors can be further defined by:

$$\underline{R} = \underline{R}_S + \underline{u}_{ZN}\, h \qquad \underline{R}_{REF} = \underline{R}_{S_{REF}} + \underline{u}_{ZREF}\, h_{REF} \tag{4.4.2.2-2}$$

where, as in Figure 5.2-1 of Chapter 5:

$\underline{R}_S$ = Position vector from the center of the earth to the point on the earth surface reference ellipsoid directly below (above) the current position location (the "navigation point"). The earth surface "reference ellipsoid" is the zero altitude reference surface defined for the earth (See the first paragraph in Section 5.1, Reference 3 - Section 4.5, and Reference 4 - Chapters 3 and 6).

h = Altitude of the navigation point above (below) the earth surface reference ellipsoid.

$\underline{u}_{ZN}$ = Unit vector upward along a line from $\underline{R}_S$ to the navigation point $\underline{R}$ (i.e., along the N Frame Z-axis). $\underline{u}_{ZN}$ is equivalent to $\underline{u}_{Up}$ in Figure 5.2-1.

$\underline{R}_{S_{REF}}$ = Position vector from the center of the earth to the point on the earth surface reference ellipsoid directly below (above) the specified earth fixed reference location point.

h_{REF} = Altitude of the selected specified earth fixed reference location point above (below) the earth surface reference ellipsoid.

$\underline{u}_{ZREF}$ = Unit vector upward along a line from $\underline{R}_{S_{REF}}$ to the specified earth fixed reference location point $\underline{R}_{REF}$. Note from the REF coordinate frame definition to follow that $\underline{u}_{ZREF}$ lies along the REF Frame Z-axis, hence, the Z tag in the $\underline{u}_{ZREF}$ subscript.

Equations (4.4.2.2-1) and (4.4.2.2-2) are now specialized to a particular REF coordinate frame where we define:

REF = Earth fixed coordinate frame with X, Y, Z axes parallel to the local East, North, Up directions at the specified earth fixed reference location point.

With this definition and the E Frame as defined in Section 4.0, Equations (4.4.2.2-1) and (4.4.2.2-2) combined with coordinate frame superscript designations become:

$$\Delta \underline{R}^{REF} = C_E^{REF}\left(\underline{R}_S^E + \underline{u}_{ZN}^E\, h - \underline{R}_{S_{REF}}^E\right) - \underline{u}_{ZREF}^{REF}\, h_{REF} \qquad (4.4.2.2\text{-}3)$$

where

C_E^{REF} = Direction cosine matrix that transforms vectors from the E to the REF Frame.

The individual matrix and vector elements in (4.4.2.2-3) can now be expressed in terms of navigation parameters. This is easily accomplished from the following Figure 4.4.2.2-1 Method of Least Work diagram obtained by expansion of Figure 4.4.2.1-2 to include the REF Frame and the $\underline{u}_{ZN}$, $\underline{u}_{ZREF}$ unit vectors from their above definitions:

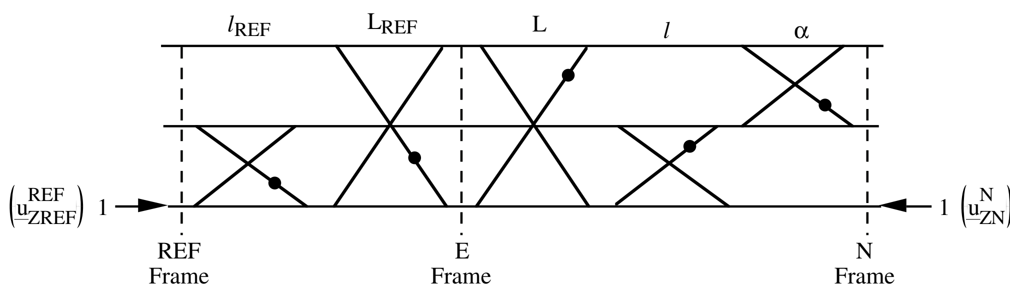


Figure 4.4.2.2-1 Coordinate Frame Relationships

where

l_{REF}, L_{REF} = Latitude, longitude of the specified earth fixed reference location point.

Using Figure 4.4.2.2-1 we write for particular terms in Equation (4.4.2.2-3):

$$\underline{u}_{ZN}^{E} = \begin{bmatrix} u_{ZN_{XE}} \\ u_{ZN_{YE}} \\ u_{ZN_{ZE}} \end{bmatrix} = \begin{bmatrix} \cos l \sin L \\ \sin l \\ \cos l \cos L \end{bmatrix} \qquad \underline{u}_{ZREF}^{REF} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\underline{u}_{ZREF}^{E} = \begin{bmatrix} u_{ZREF_{XE}} \\ u_{ZREF_{YE}} \\ u_{ZREF_{ZE}} \end{bmatrix} = \begin{bmatrix} \sin L_{REF} \cos l_{REF} \\ \sin l_{REF} \\ \cos L_{REF} \cos l_{REF} \end{bmatrix} \tag{4.4.2.2-4}$$

$$C_{E}^{REF} = \begin{bmatrix} \cos L_{REF} & 0 & -\sin L_{REF} \\ -\sin L_{REF} \sin l_{REF} & \cos l_{REF} & -\cos L_{REF} \sin l_{REF} \\ \sin L_{REF} \cos l_{REF} & \sin l_{REF} & \cos L_{REF} \cos l_{REF} \end{bmatrix}$$

where

$u_{ZN_{XE}}, u_{ZN_{YE}}, u_{ZN_{ZE}}$ = X, Y, Z components of $\underline{u}_{ZN}^{E}$.

$u_{ZREF_{XE}}, u_{ZREF_{YE}}, u_{ZREF_{ZE}}$ = X, Y, Z components of $\underline{u}_{ZREF}^{E}$.

The remaining terms in Equation (4.4.2.2-3) are from Equations (5.1-9) and (5.1-10) in Section 5.1 (recognizing $\underline{u}_{Up}$ as the equivalent to the unit vectors along the Z axes of the N and REF Frames):

$$\underline{R}_{S}^{E} = R_{S}' \begin{bmatrix} u_{ZN_{XE}} \\ (1 - e)^2 \, u_{ZN_{YE}} \\ u_{ZN_{ZE}} \end{bmatrix} \qquad R_{S}' = R_0 / \sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^2 - 1 \right]}$$

$$\tag{4.4.2.2-5}$$

$$\underline{R}_{S_{REF}}^{E} = R_{S_{REF}}' \begin{bmatrix} u_{ZREF_{XE}} \\ (1 - e)^2 \, u_{ZREF_{YE}} \\ u_{ZREF_{ZE}} \end{bmatrix} \qquad R_{S_{REF}}' = R_0 / \sqrt{1 + u_{ZREF_{YE}}^2 \left[(1 - e)^2 - 1 \right]}$$

where

R_0 = Earth's equatorial radius.

e = Earth surface reference ellipsoid oblateness (ellipticity) coefficient.

Finally, it is also useful to obtain an expression for the C_N^{REF} direction cosine matrix for transforming vector data (such as velocity) from the navigation N Frame (in which it is calculated) to the selected earth fixed REF coordinate frame. The C_N^{REF} matrix is easily computed using the Equation (3.2.1-5) direction cosine matrix chain rule:

$$C_N^{REF} = C_E^{REF} \, C_N^E \tag{4.4.2.2-6}$$

with C_E^{REF} as calculated in Equations (4.4.2.2-4).

In summary, Equation (4.4.2.2-3) with (4.4.2.2-4) and (4.4.2.2-5) allow calculation of the position vector from a specified earth fixed reference point to the current position location as a function of current navigation point latitude, longitude, altitude and the specified reference point latitude, longitude, altitude. The position vector is calculated in a reference coordinate REF Frame that is parallel to North, East, Up axes at the specified reference position point. Equation (4.4.2.2-6) defines the C_N^{REF} direction cosine matrix for transforming vector data from the current navigation point N Frame to the selected reference coordinate Frame REF.

4.4.2.3 LATITUDE, LONGITUDE, ALTITUDE FROM POSITION VECTOR IN ARBITRARY EARTH FIXED COORDINATE FRAME

For situations when position is represented as a position vector $\Delta\underline{R}$ from a specified reference position location to the current position location, the corresponding latitude, longitude, altitude can be derived by first writing the following equivalent form of Equation (4.4.2.2-1):

$$\underline{R} = \underline{R}_{REF} + \Delta\underline{R} \tag{4.4.2.3-1}$$

Substituting from Equations (4.4.2.2-2) for $\underline{R}_{REF}$ and specializing to earth E Frame coordinates then obtains:

$$\underline{R}^E = \underline{R}^E_{SREF} + \underline{u}^E_{ZREF} \, h_{REF} + C^E_{REF} \, \Delta\underline{R}^{REF} \tag{4.4.2.3-2}$$

in which the REF Frame is as defined in Section 4.4.2.2. The contributing terms in Equation (4.4.2.3-2) have been presented in Equations (4.4.2.2-4) and (4.4.2.2-5), recognizing from generalized Equation (3.2.1-3) that C_{REF}^{E} is the transpose of C_{E}^{REF}.

Equation (4.4.2.3-2) with (4.4.2.2-4) and (4.4.2.2-5) allows the position vector from earth's center to be determined in E Frame coordinates ($\underline{R}^E$) from the defined $\Delta\underline{R}^{REF}$ position vector and the selected reference point latitude, longitude, altitude. Latitude, longitude, altitude of the current position location is then determined from $\underline{R}^E$ as follows.

First longitude is extracted. From Equations (4.4.2.2-4) and (5.2.2-1) (with $\underline{u}_{ZN}$ for $\underline{u}_{Up}$) we note that:

$$\underline{u}_{ZN}^{E} = \begin{bmatrix} u_{ZN_{XE}} \\ u_{ZN_{YE}} \\ u_{ZN_{ZE}} \end{bmatrix} = \begin{bmatrix} \cos l \sin L \\ \sin l \\ \cos l \cos L \end{bmatrix} \tag{4.4.2.3-3}$$

$$\frac{u_{ZN_{XE}}}{u_{ZN_{ZE}}} = \frac{R_{XE}}{R_{ZE}} \tag{4.4.2.3-4}$$

where

R_{XE}, R_{ZE} = X, Z components of $\underline{R}^E$.

From Equation (4.4.2.3-3) we see that longitude can be calculated as:

$$L = \tan^{-1} \frac{u_{ZN_{XE}}}{u_{ZN_{ZE}}} \tag{4.4.2.3-5}$$

Substituting (4.4.2.3-4) into (4.4.2.3-5) then yields the longitude extraction formula:

$$L = \tan^{-1} \frac{R_{XE}}{R_{ZE}} \tag{4.4.2.3-6}$$

Extraction of latitude and altitude is performed concurrently through solution of the following simultaneous equations derived from Equations (5.1-10) and (5.2.1-4), the quadratic solution to (5.2.1-5), the Y component of (5.2.2-1), and (4.4.2.3-3), with $\underline{u}_{ZN}$ for $\underline{u}_{Up}$:

$$\begin{aligned} R_S' &= R_0 / \sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \\ R_S &= R_S' \sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^4 - 1 \right]} \end{aligned} \tag{4.4.2.3-7}$$

(Continued)

$$R^2 = \underline{R}^E \cdot \underline{R}^E$$

$$h = -\frac{R_0^2}{R_S'} + \sqrt{\left(\frac{R_0^2}{R_S'}\right)^2 + R^2 - R_S^2} \tag{4.4.2.3-7 (Continued)}$$

$$u_{ZN_{YE}} = R_{YE} \,/\, \left[(1 - e)^2 R_S' + h\right]$$

$$l = \tan^{-1} \frac{u_{ZN_{YE}}}{\sqrt{1 - u_{ZN_{YE}}^2}} \tag{4.4.2.3-8}$$

where

R_{YE} = Y component of $\underline{R}^E$.

Note that the positive sign has been selected for the h quadratic solution formula in (4.4.2.3-7) to correspond with the realistic solution on "this side of the earth" (the negative sign solution corresponds to a local vertical on the opposite side of the earth).

Equations (4.4.2.3-7) are first solved for altitude h and $u_{ZN_{YE}}$. The $u_{ZN_{YE}}$ solution is then used to determine latitude through Equation (4.4.2.3-8). Due to the complexity of Equations (4.4.2.3-7) it has not been possible to condense them into a closed-form exact analytic solution for h and $u_{ZN_{YE}}$. A precise numerical solution is readily achievable, however, by successive iteration of Equations (4.4.2.3-7) using the $u_{ZN_{YE}}$ value obtained during the last iteration. A starting value of $u_{ZN_{YE}} = 0$ suffices for the iteration process. Three to five iterations is usually adequate to achieve an altitude h and latitude l solution that meets the most stringent accuracy requirements. The number of iteration cycles actually used should be based on simulation analyses.

4.4.3 LATITUDE/LONGITUDE CALCULATED BY DIRECT INTEGRATION

The rate of change of latitude, longitude and wander angle can be written from The Method of Least Work diagram in Figure 4.4.3-1, obtained from Figure 4.4.2.1-2 by adding the Euler angle rate terms (See Section 3.3.3.3 for review):

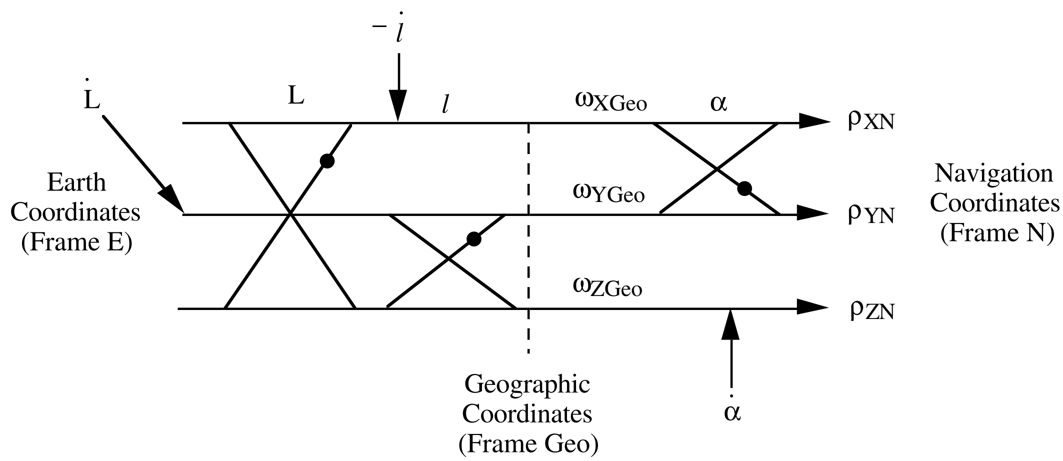


Figure 4.4.3-1 Latitude, Longitude, Wander Angle Rate Description

where

$\rho_{XN}, \rho_{YN}, \rho_{ZN}$ = N Frame components of the N Frame rotation rate relative to the earth (E) as defined by the "transport rate" $\underline{\rho}^N$ in Equations (4.1.1-6).

$\omega_{XGeo}, \omega_{YGeo}, \omega_{ZGeo}$ = Geo (East, North, Up) Frame X, Y, Z components of Geo Frame angular rate relative to the E Frame.

Figure 4.4.3-1 differs from Figure 3.3.3.3-1 in not containing additional $\underline{\omega}$ angular rate vector component inputs on the left side of the diagram (i.e., the $\omega_{XA}, \omega_{YA}, \omega_{ZA}$ terms in Figure 3.3.3.3-1). In Figure 4.4.3-1 these would represent the angular rate of the E Frame. Since the $\rho_{XN}, \rho_{YN}, \rho_{ZN}$ outputs represent the angular rate of the N Frame relative to the E Frame, $\omega_{XA}, \omega_{YA}, \omega_{ZA}$ components input from the left would represent the angular rate of the E Frame relative to the E Frame which is clearly zero. Also, note in Figure 4.4.3-1 that the latitude rate input is negative because the latitude Euler angle is defined as an Euler angle rotation about the negative X-axis (see discussion following Figure 3.3.3.3-1).

From Figure 4.4.3-1 we can write for the Geo Frame angular rate components:

$$\omega_{XGeo} = -\dot{l} \qquad \omega_{YGeo} = \dot{L}\cos l \qquad \omega_{ZGeo} = \dot{L}\sin l \qquad (4.4.3\text{-}1)$$

Equations (4.4.3-1) can be rearranged to obtain expressions for latitude, longitude rate as a function of horizontal Geo Frame angular rate components:

$$\dot{l} = -\,\omega_{XGeo} \qquad \dot{L} = \omega_{YGeo} \sec\, l \tag{4.4.3-2}$$

If Equations (4.4.3-2) are utilized to calculate latitude, longitude, it is usually accompanied by use of a “geographic” local level navigation N Frame configuration in which the X and Y axes are controlled to lie East and North (i.e., along Geo axes). To create an N Frame that is aligned with Geo axes (i.e., “North slaved”), the wander angle α must be initialized at zero, and the wander angle rate $\dot{\alpha}$ must be held at zero. From Figure 4.4.3-1 and Equations (4.4.3-1) - (4.4.3-2) we see that the N Frame Z axis transport rate is given by:

$$\rho_{ZN} = \omega_{ZGeo} + \dot{\alpha} = \dot{L} \sin l + \dot{\alpha} = \omega_{YGeo} \tan\, l + \dot{\alpha} \tag{4.4.3-3}$$

from which:

$$\dot{\alpha} = \rho_{ZN} - \omega_{YGeo} \tan\, l \tag{4.4.3-4}$$

Equation (4.4.3-4) shows that the wander angle rate can be maintained at zero by setting the N Frame vertical transport rate component to:

$$\rho_{ZN} = \omega_{YGeo} \tan\, l \tag{4.4.3-5}$$

For zero initial wander angle, the N Frame horizontal transport rate components then become:

$$\rho_{XN} = \omega_{XGeo} \qquad \rho_{YN} = \omega_{YGeo} \tag{4.4.3-6}$$

For zero wander angle, the ω_{XGeo}, ω_{YGeo} components in (4.4.3-2), (4.4.3-5) and (4.4.3-6) are the X, Y components of Equations (5.3-17) - (5.3-18) with (5.1-10), (5.2.3-1), (5.2.4-25) and (5.2.4-37) using (4.4.2.1-2) for D_{21}, D_{22} and D_{23} with $\alpha = 0$:

$$\begin{gathered}
\omega_{XGeo} = -\frac{1}{r_l} v_{YN} \qquad \omega_{YGeo} = \frac{1}{r_l}\left(1 + f_{eh} \cos^2 l\right) v_{XN} \\
r_l = r_{ls} + h \qquad r_{ls} = \frac{R_S'^{\,3}}{R_0^2}\left(1 - e^2\right) \\
f_{eh} = \frac{(1 - e)^2 - 1}{\left(1 + \left[(1 - e)^2 - 1\right] \sin^2 l\right)} \frac{1}{\left(1 + h / R_S'\right)} \\
R_S' = R_0 / \sqrt{1 + \left[(1 - e)^2 - 1\right] \sin^2 l}
\end{gathered} \tag{4.4.3-7}$$

where

v_{XN}, v_{YN} = X, Y components of $\underline{v}^N$.

Using (4.4.3-7), Equations (4.4.3-2) can be integrated to determine latitude and longitude directly. Note, however, that the secant latitude term in the longitude rate equation becomes infinite at the polar regions ($l = \pm \pi / 2$). As a result, Equations (4.4.3-2) cannot be utilized for all global positions, which has generally made them unpopular for application in modern day inertial navigation software implementations.

Equation (4.4.3-5) shows that a similar problem arises in the N Frame vertical component of transport rate due to the tangent latitude term, which also becomes infinite in the polar regions. The result is that $\underline{\omega}_{EN}^N$ (the formal definition for $\underline{\rho}^N$) becomes infinite, generating velocity rate infinities in Equation (4.3-18) and attitude rate infinities in Equations (4.1-1) and (4.1-2) from Equation (4.1.1-1).

4.5 LOCAL LEVEL COORDINATE FRAME N OPTIONS

By its definition in Section 2.2, navigation coordinate Frame N has the Z axis up and the X, Y axes horizontal. As discussed in Section 4.4.1.1, the orientation of the X, Y axes around Z is somewhat arbitrary, and depends on the selection of the vertical component of transport rate ρ_{ZN} used in the C_N^E direction cosine matrix rate Equation (4.4.1.1-1).

An initial choice for ρ_{ZN} might be to rotate the navigation axes to maintain a parallel alignment with local North/East geographic axes (i.e., the Geo Frame of Section 2.2). Vector components calculated in Frame N (such as the Section 4.3.1 velocity) would then automatically lie along North/East/Vertical axes, a typically desired form for system output. Additionally, heading data defined by the heading attitude of body Frame B relative to the locally level attitude reference Frame L (hence, Frame N via Equation (4.1.1-2)) would be referenced to North, another desirable feature for output. Equations (4.4.3-5) and (4.4.3-6) of Section 4.4.3 show that the required value of ρ_{ZN} for a geographic navigation N Frame is given by:

$$\rho_{ZN} = \rho_{YN} \tan l \tag{4.5-1}$$

Equation (4.5-1) reveals that for a geographic N Frame, a singularity exists near the earth poles ($l = \pm 90°$) in the vertical component of $\underline{\rho}^N$. Thus, use of such a system must be restricted to travel away from the poles to avoid introducing large errors in the N Frame rotation rate, which, as discussed in Section 4.4.3, would introduce large errors in the attitude, velocity and position rate integrations that all utilize $\underline{\rho}^N$.

If we arbitrarily set ρ_{ZN} to zero, the N Frame implementation is denoted as a "wander azimuth" configuration. With a wander azimuth N Frame ($\rho_{ZN} = 0$), we see from Equation (4.4.3-3) that the α wander angle rate of change is zero for a stationary vehicle (i. e., under zero longitude L rate of change). For this condition, the wander angle remains constant. Under translational motion relative to the earth (i.e., Easterly movement which changes longitude), the wander angle (i.e., the azimuth orientation of the N Frame from North) develops a rate of change, hence the terminology "wander azimuth".

If ρ_{ZN} is set equal to the negative of the vertical earth rate component relative to inertial space, a free azimuth N Frame implementation would result. For a free azimuth N Frame, the vertical angular rate of the attitude reference coordinate Frame L relative to inertial space (Frame I) is zero (see Equation (4.1.1-1) with (4.1.1-2)), hence, the terminology "free azimuth"; i.e., letting it "run inertially free".

For either the wander azimuth or free azimuth approach, ρ_{ZN} is finite by definition for all earth position locations. Since the horizontal components are also finite (as discussed in Section 4.4.1.1) no singularities exist for $\underline{\rho}^N$, and the local level navigation frame rotation rate is completely defined for all earth trajectories. The $\underline{\rho}^N$ generated singularity conditions (in attitude, velocity and position rates) associated with the geographic local level N Frame approach is, thereby, avoided.

The wander azimuth implementation ($\rho_{ZN} = 0$) is the method usually selected for strapdown inertial navigation systems for which Equations (4.4.1.1-3) and (4.1.1-7) assume the simplified form:

$$
\begin{aligned}
\dot{D}_{11} &= - D_{13}\, \rho_{YN} \\
\dot{D}_{12} &= D_{13}\, \rho_{XN} \\
\dot{D}_{13} &= D_{11}\, \rho_{YN} - D_{12}\, \rho_{XN} \\
\\
\dot{D}_{21} &= - D_{23}\, \rho_{YN} \\
\dot{D}_{22} &= D_{23}\, \rho_{XN} \\
\dot{D}_{23} &= D_{21}\, \rho_{YN} - D_{22}\, \rho_{XN}
\end{aligned}
\qquad (4.5\text{-}2)
$$

(Continued)

$$
\begin{aligned}
\dot{D}_{31} &= - D_{33}\, \rho_{YN} \\
\dot{D}_{32} &= D_{33}\, \rho_{XN} \\
\dot{D}_{33} &= D_{31}\, \rho_{YN} - D_{32}\, \rho_{XN}
\end{aligned}
\tag{4.5-2 (Continued)}
$$

$$
\begin{aligned}
\omega_{XL} &= D_{22}\, \omega_e + \rho_{YN} \\
\omega_{YL} &= D_{21}\, \omega_e + \rho_{XN} \\
\omega_{ZL} &= - D_{23}\, \omega_e
\end{aligned}
\tag{4.5-3}
$$

4.6 INITIALIZATION

The fundamental strapdown inertial navigation computational operation is an integration process to determine attitude, velocity and position using sensed angular rate (from strapdown inertial angular rate sensors) and linear specific force acceleration (from strapdown accelerometers). Before the integration process can be initiated, values must be assigned to the navigation integration parameters which from Sections 4.1, 4.3, and 4.4 are typically C_B^L (or q_B^L), C_N^E, $\underline{v}^N$, and h. The initialization process is embodied in a set of dynamic equations that are processed over an initialization period usually referred to as “Initial Alignment”.

For most applications, initialization of position and velocity during Initial Alignment utilizes position, velocity input data provided from another source of navigational information (an exception is in the case of quasi-stationary initial alignment discussed in Chapter 6 in which velocity inputs are not explicitly required, but are assumed to be zero on the average based on the quasi-stationary environment). The attitude (verticality and heading) initialization process, is generally required to be accomplished without the benefit of external attitude inputs (also with some exceptions, e.g., satellite inertial attitude initialization based on stellar sightings). The general attitude initialization process is typically based on an analytic implementation of the strapdown attitude update and acceleration transformation operations such as calculated (using direction cosine attitude parameters) by Equations (4.2-1) and (4.2-3) with (4.1-1) and (4.1.1-1) repeated below:

$$
\begin{aligned}
\dot{C}_B^L &= C_B^L \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IL}^L \times\right) C_B^L \\
\underline{\omega}_{IL}^L &= C_N^L \left(\underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N\right) \\
\underline{a}_{SF}^L &= C_B^L\, \underline{a}_{SF}^B \\
\underline{a}_{SF}^N &= C_L^N\, \underline{a}_{SF}^L
\end{aligned}
\tag{4.6-1}
$$

The initialization of attitude as represented by the C_B^L matrix is accomplished through a dynamic estimation/control loop, built analytically around Equations (4.6-1), that determines inaccuracies in C_B^L through observation of their effects on the N Frame specific force acceleration vector $\underline{a}_{SF}^N$ calculated from the $\underline{a}_{SF}^B$ transformation operation. Observed effects on $\underline{a}_{SF}^N$ are filtered and fed back through a dynamic servo loop process to continually correct and refine the accuracy of C_B^L.

For applications in which the strapdown system being initialized has motion relative to the earth ("moving base alignment"), the specific force acceleration $\underline{a}_{SF}^B$ can have dynamic components that amplify the effects of C_B^L error on $\underline{a}_{SF}^N$. In some of these applications, intentional maneuvering is performed to accentuate the effect (the so called "transfer alignment" procedure). In many applications, Initial Alignment is performed at a "quasi-stationary" condition in which the strapdown inertial system is essentially stationary but may have small bounded random position and attitude disturbances (such as in a parked airplane in the presence of wind gusts while undergoing fuel, stores, crew, and passenger loading).

In the quasi-stationary situation, the specific force acceleration vector is essentially vertical, hence, only amplifies variations in C_B^L from vertical (i.e., "tilt"). For such a case, the C_B^L heading is determined through observation of how errors in the N Frame earth rate term $\underline{\omega}_{IE}^N$ affect $\underline{a}_{SF}^N$ through its impact on $\underline{\omega}_{IL}^L$, hence C_B^L, in the $\dot{C}_B^L$ equation. Observed effects of $\underline{\omega}_{IE}^N$ on $\underline{a}_{SF}^N$ are filtered and fed back as part of the dynamic servo loop Initial Alignment process to continually correct and refine an estimate for $\underline{\omega}_{IE}^N$. Once the horizontal components of $\underline{\omega}_{IE}^N$ are determined to sufficient accuracy, the heading attitude of the N Frame relative to the earth is established from the knowledge that the horizontal earth rate projection points north. Since the C_N^L matrix is a known constant from Equation (4.1.1-2), this also defines the initial heading attitude of the L Frame.

The previous over-simplified discussion should be viewed only as a very brief introduction to fundamental initialization requirements and procedures. A detailed analytical description of quasi-stationary Initial Alignment is presented in Chapter 6. Both quasi-stationary and moving base Initial Alignment are discussed in further detail in Sections 15.2.1 and 15.2.2 of Chapter 15.

4.7 STRAPDOWN INERTIAL NAVIGATION EQUATION SUMMARY

Table 4.7-1 is a listing of the principal equations from Chapter 4 (including Chapter 5 earth related parameters and Chapter 6 quasi-stationary initialization) typically utilized in strapdown inertial navigation system software packages. Table 4.7-1 lists the equation function, input parameters, output parameters and equation number. Definitions for the input/output parameters can be found by cross-referencing through the Parameter Index in the back of the book.

Table 4.7-1 Summary Of Typical Strapdown Inertial Navigation System Equations

EQUATION FUNCTION	INPUT	OUTPUT	EQUATION
Quasi-Stationary Initialization	Latitude, Longitude & Altitude	C_B^L (or q_B^L), $\underline{v}^N$, C_N^E, h	Chapt. 6
Earth Related Parameters	Earth Shape, Gravity & Angular Rate Constants, $\underline{v}^N$, C_N^E, h	$\underline{\rho}^N = \underline{\omega}_{EN}^N$, $\underline{g}_P^N$	Table 5.6-1
N To L Frame Direction Cosine Matrix (Constant)	Definition	$C_N^L = \left(C_L^N\right)^T$	(4.1.1-2)
N Frame Earth Rate Vector	C_N^E, ω_e	$\underline{\omega}_{IE}^N$	(4.1.1-3) & (4.1.1-4)
L Frame Angular Rate Vector	$\underline{\omega}_{IE}^N$, $\underline{\omega}_{EN}^N$, C_N^L	$\underline{\omega}_{IL}^L$	(4.1.1-1)
Acceleration Transformation	$\underline{a}_{SF}^B$, C_B^L, C_L^N	$\underline{a}_{SF}^N$	(4.2-1) & (4.2-3)
Vertical Channel Control Gains	Constants	C_1, C_2, C_3	(4.4.1.2.1-11)
Vertical Channel Control Terms	C_1, C_2, C_3, h, h_{Prsr}	e_{vc_1}, e_{vc_2}	(4.4.1.2.1-3)
Unit Vector Upward In N Frame (Constant)	Definition	$\underline{u}_{ZN}^N$	(5.3-18)

EQUATION FUNCTION	INPUT	OUTPUT	EQUATION
Velocity Differential Equation To Be Integrated	$\underline{a}_{SF}^{N}$, $\underline{g}_{P}^{N}$, $\underline{\omega}_{EN}^{N}$, $\underline{\omega}_{IE}^{N}$, $\underline{v}^{N}$, e_{vc_1}, $\underline{u}_{ZN}^{N}$	$\underline{v}^{N}$	(4.4.1.2.1-1)
East, North, Up Velocity Component Outputs	$\underline{v}^{N}$, α	v_{East}, v_{North}, v_{Up}	(4.3.1-4)
Altitude Differential Equation To Be Integrated	$\underline{v}^{N}$, $\underline{u}_{ZN}^{N}$, e_{vc_2}	h	(4.4.1.2.1-2)
Position Direction Cosine Matrix Differential Equation To Be Integrated	$\underline{\rho}^{N}$, C_{N}^{E}	C_{N}^{E}	(4.4.1.1-1)
Latitude, Longitude Outputs And Wander Angle	C_{N}^{E}	l, L, α	(4.4.2.1-3)
Attitude Differential Equation To Be Integrated (Direction Cosine Matrix Form)	$\underline{\omega}_{IB}^{B}$, $\underline{\omega}_{IL}^{L}$, C_{B}^{L}	C_{B}^{L}	(4.1-1)
Attitude Differential Equation To Be Integrated (Quaternion Form)	$\underline{\omega}_{IB}^{B}$, $\underline{\omega}_{IL}^{L}$, q_{B}^{L}	q_{B}^{L}	(4.1-2) & (4.1-3)
Attitude Quaternion To Attitude Direction Cosine Matrix Conversion (For Attitude Quaternion As Basic Attitude Form)	q_{B}^{L}	C_{B}^{L}	(4.2-5)
Roll, Pitch, True Heading Euler Angle Outputs	C_{B}^{L}, α	ϕ, θ, ψ_{True}	(4.1.2-1) & (4.1.2-2)

5 Earth Related Navigation Parameters

5.0 OVERVIEW

In Chapter 4 we used plumb-bob gravity ($\underline{g}_P^N$), transport rate ($\underline{\rho}^N$) and analytical relationships between C_N^E, h and other position location representations to describe velocity rate ($\dot{\underline{v}}^N$) and various position determination approaches. Each of these parameters is related to the analytical model used to describe gravity and the shape of the earth's surface. In this chapter, we will develop analytically exact equations for gravity, earth's surface shape, and other earth related terms based on the standard ellipsoidal earth surface shape and gravity model of Reference 4 (also discussed in Reference 3 - Chapter 4). The coordinate frames we will be utilizing in this chapter will be Frames E, N and Geo (defined in Section 2.2), where the E Frame axes are specified in particular to be:

E = Earth fixed coordinate frame having its Y axis along the earth's polar axis with X and Z axes in the equatorial plane. The E Frame Z Axis is defined to lie along the intersection of the Greenwich reference meridian plane (contains Greenwich England) and the equatorial plane.

A table is provided at the end of this chapter listing the salient equations developed in the order they would be applied in the strapdown INS computer.

5.1 EARTH SHAPE MODEL

For navigational purposes, the shape of the earth's reference surface is approximated as an ellipsoid of revolution around the earth polar axis. This imaginary reference ellipsoid is the surface from which "altitude" is measured and is defined as the ellipsoid that "best fits" a reference geoid (a surface of constant gravity field potential) selected to approximate mean sea level. (Note; gravitational acceleration is the gradient of the gravity potential field - See Reference 3, Section 4.4 for further explanation). When we speak of the altitude of a position location over the earth, we mean the height above the reference ellipsoid measured along a line that passes through the position point and is perpendicular to the surface of the reference ellipsoid below (or above) the position location. By "geodetic vertical", we mean the direction of this "line".

The equation for points on the surface of the earth reference ellipsoid can be described in the earth fixed E Frame by the surface function:

$$\xi \;=\; R_{S_{XE}}^{2} + R_{S_{ZE}}^{2} + \frac{R_{S_{YE}}^{2}}{(1 - e)^{2}} - R_{0}^{2} \;=\; 0 \tag{5.1-1}$$

in which the symmetric axis of the ellipsoid (i.e., earth's polar axis) is parallel to the E Frame Y axis, and where:

$\underline{R}_S$ = Distance vector from the center of the ellipsoid to a point on the ellipsoid surface.

$R_{S_{XE}}, R_{S_{YE}}, R_{S_{ZE}}$ = X, Y, Z components of $\underline{R}_S^E$, the projection of $\underline{R}_S$ on E Frame axes.

R_0 = Semi-major axis of the reference ellipse (i.e., earth's equatorial radius). The numerical value from Reference 4 (Table 3.1) is provided in Table 5.6-1 at the conclusion of this chapter.

e = Ellipticity of the reference ellipse defined as one minus the ratio of the semi-minor over the semi-major axis (also known as "flattening"). The numerical value from Reference 4 (Table 3.1) is provided in Table 5.6-1 at the conclusion of this chapter.

ξ = Ellipsoid surface function.

Figure 5.1-l illustrates the geometry involved in which the earth's ellipticity has been intentionally exaggerated from its true much smaller value.

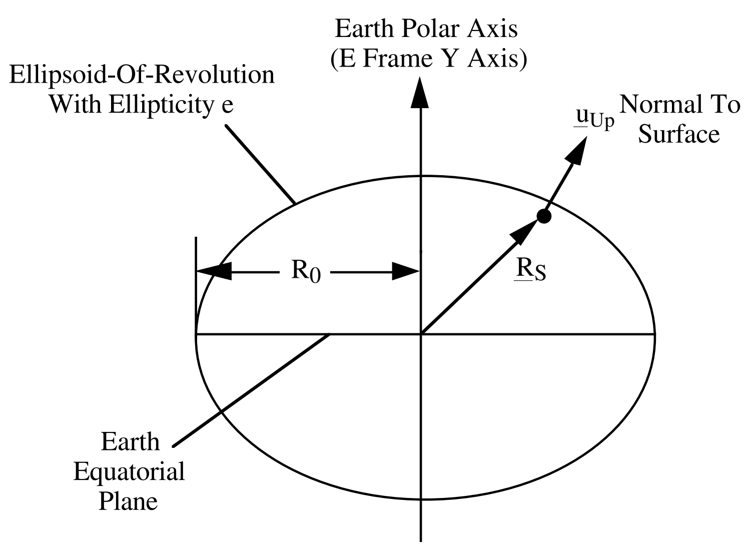


Figure 5.1-1 Earth Surface Analytical Model

Now imagine a unit vector perpendicular to the ellipsoid surface at the $\underline{R}_S$ surface point (i.e., along the geodetic vertical). If a particular position location is along the unit vector directly above (or below) the $\underline{R}_S$ surface point, the unit vector would lie along the Z axis of the Section 2.2 defined N, L and Geo coordinate frames corresponding to that position location. Figure 5.1-1 illustrates such a unit vector labeled $\underline{u}_{Up}$, where:

$\underline{u}_{Up}$ = Unit vector perpendicular to the ellipsoid surface at $\underline{R}_S$, corresponding to a position location directly above (or below) $\underline{R}_S$ along $\underline{u}_{Up}$.

An analytical expression for $\underline{u}_{Up}$ can be derived from the property that a vector normal to a surface function ξ lies along the surface function gradient (Reference 37 - Chapter 5, Section 3), hence, in the E Frame we can write:

$$\underline{u}_{Up}^{E} = \frac{\nabla \xi}{\left| \nabla \xi \right|} \tag{5.1-2}$$

where

∇ = E Frame vector gradient operator.

$\underline{u}_{Up}^{E}$ = $\underline{u}_{Up}$ projected on E Frame axes.

By definition of the gradient (Reference 37 - Chapter 5, Section 3) applied in the E Frame:

$$\nabla \xi = \left[\frac{\partial \xi}{\partial R_{S_{XE}}} , \frac{\partial \xi}{\partial R_{S_{YE}}} , \frac{\partial \xi}{\partial R_{S_{ZE}}} \right]^{T} \tag{5.1-3}$$

With (5.1-1) in Equation (5.1-3) we find:

$$\nabla \xi = \left[2\, R_{S_{XE}}, 2 \frac{R_{S_{YE}}}{(1 - e)^{2}}, 2\, R_{S_{ZE}} \right]^{T} \tag{5.1-4}$$

$$\left| \nabla \xi \right| = 2 \sqrt{\left(R_{S_{XE}}^{2} + R_{S_{ZE}}^{2} + R_{S_{YE}}^{2} / (1 - e)^{4} \right)} \tag{5.1-5}$$

For convenience, we define $R_S^{\prime}$ (not to be confused with R_S, the magnitude of $\underline{R}_S$, to be introduced subsequently):

$$R_S^{\prime} \equiv \sqrt{R_{S_{XE}}^{2} + R_{S_{ZE}}^{2} + R_{S_{YE}}^{2} / (1 - e)^{4}} \tag{5.1-6}$$

where

$R_S^{\prime}$ = Modified magnitude of $\underline{R}_S$.

so that:

$$\left| \nabla \xi \right| = 2\, R_S' \tag{5.1-7}$$

Substituting (5.1-4) and (5.1-7) into (5.1-2) obtains for $\underline{u}_{Up}^E$ in terms of $\underline{R}_S^E$ components:

$$\begin{aligned} u_{Up_{XE}} &= R_{S_{XE}} / R_S' \\ u_{Up_{YE}} &= R_{S_{YE}} / (1 - e)^2\, R_S' \\ u_{Up_{ZE}} &= R_{S_{ZE}} / R_S' \end{aligned} \tag{5.1-8}$$

where

$u_{Up_{XE}}, u_{Up_{YE}}, u_{Up_{ZE}}$ = E Frame X, Y, Z components of $\underline{u}_{Up}$.

or, upon inversion, for $\underline{R}_S^E$ in terms of $\underline{u}_{Up}^E$ components:

$$\begin{aligned} R_{S_{XE}} &= u_{Up_{XE}}\, R_S' \\ R_{S_{YE}} &= (1 - e)^2\, u_{Up_{YE}}\, R_S' \\ R_{S_{ZE}} &= u_{Up_{ZE}}\, R_S' \end{aligned} \tag{5.1-9}$$

The $R_{S_{YE}}$ expression in (5.1-9) can be used to derive an alternate to (5.1-6) for R_S'. We first combine (5.1-l) with (5.1-6) squared and expand:

$$R_S'^{\,2} = R_0^2 - R_{S_{YE}}^2 / (1 - e)^2 + R_{S_{YE}}^2 / (1 - e)^4 = R_0^2 + \left[R_{S_{YE}}^2 / (1 - e)^4 \right] \left[1 - (1 - e)^2 \right]$$

Substituting $R_{S_{YE}}$ from (5.1-9) and rearranging then gives:

$$R_S' = R_0 / \sqrt{1 + u_{Up_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \tag{5.1-10}$$

5.2 ELLIPSOIDAL EARTH REFERENCED NAVIGATION PARAMETERS

As part of the inertial navigation computational process, earth referenced parameters must be calculated based on the ellipsoidal earth model, under conditions when the navigating vehicle is at a position above (or below) the earth surface. For an intentionally exaggerated earth ellipticity (for illustrative purposes), Figure 5.2-1 describes the relative position geometry and the definition of position variables used in subsequent analytic developments.

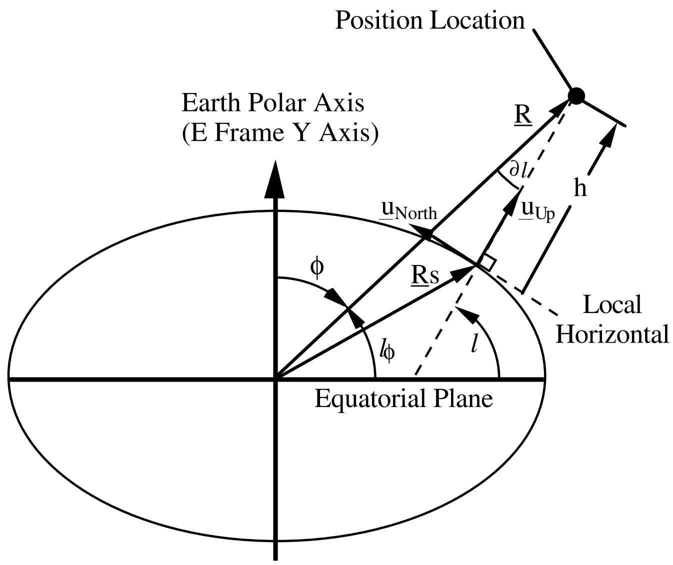


Figure 5.2-1 Position Relative To Earth In Local Meridian Plane

Figure 5.2-1 describes a position location above the earth surface showing the previously defined earth surface and normal vectors ($\underline{R}_S$ and $\underline{u}_{Up}$), and where:

$\underline{R}$ = Position vector from the center of the earth to the current actual position location.

$\underline{u}_{North}$ = Unit vector in the horizontal North direction.

h = Altitude above the earth reference ellipsoid measured from earth's surface along the earth surface normal unit vector $\underline{u}_{Up}$ to the current position location.

l = Geodetic latitude defined as the angle from earth's equatorial plane to $\underline{u}_{Up}$ (positive for position locations in the Northern hemisphere).

l_ϕ = Geocentric latitude defined as the angle from earth's equatorial plane to $\underline{R}$ (positive for position locations in the Northern hemisphere).

∂l = Difference between geodetic and geocentric latitudes, known as the "deflection of the vertical".

ϕ = Angle from earth's positive polar axis (the E Frame Y axis) to $\underline{R}$ which is also the complement of l_ϕ. The angle ϕ lies in the range of zero to π, depending on the position location.

Not shown in Figure 5.2-1, but also of interest, are radii of curvature of the earth's surface at the earth surface position location, both in the plane of the local meridian, and perpendicular to the local meridian plane. The equivalent radii of curvature at the actual position location (i.e., at altitude) must also be defined.

5.2.1 MAGNITUDES OF R AND R_S

Equations for the magnitudes of $\underline{R}$ and $\underline{R}_S$ can be derived by first writing from Figure 5.2.1-1:

$$\underline{R} = \underline{R}_S + h\,\underline{u}_{Up} \tag{5.2.1-1}$$

The magnitude squared of $\underline{R}$ is then obtained from (5.2.1-1) as:

$$\begin{aligned} R^2 &= \underline{R} \cdot \underline{R} = \underline{R}_S \cdot \underline{R}_S + 2\,h\,\underline{u}_{Up} \cdot \underline{R}_S + h^2 \\ &= R_S^2 + h^2 + 2\,h\,\underline{u}_{Up} \cdot \underline{R}_S \end{aligned} \tag{5.2.1-2}$$

where

R, R_S = Magnitudes of $\underline{R}$ and $\underline{R}_S$.

Using Section 5.1 definitions for the E Frame components of $\underline{R}_S$, the magnitude squared of $\underline{R}_S$:

$$R_S^2 = R_{S_{XE}}^2 + R_{S_{YE}}^2 + R_{S_{ZE}}^2$$

which, with (5.1-9) from Section 5.1, becomes:

$$\begin{aligned} R_S^2 &= R'^2_S \left[u_{UpXE}^2 + u_{UpYE}^2 (1 - e)^4 + u_{UpZE}^2 \right] \\ &= R'^2_S \left[u_{UpXE}^2 + u_{UpYE}^2 + u_{UpZE}^2 + u_{UpYE}^2 \left[(1 - e)^4 - 1 \right] \right] \end{aligned} \tag{5.2.1-3}$$

Since $\underline{u}_{Up}$ is a unit vector, we then find for R_S:

$$R_S = R'_S \sqrt{1 + u_{UpYE}^2 \left[(1 - e)^4 - 1 \right]} \tag{5.2.1-4}$$

Using (5.1-9) for the E Frame components of $\underline{R}_S$, the dot product in (5.2.1-2) is:

$$\begin{aligned} \underline{u}_{Up} \cdot \underline{R}_S &= u_{UpXE}\,R_{S_{XE}} + u_{UpYE}\,R_{S_{YE}} + u_{UpZE}\,R_{S_{ZE}} \\ &= R'_S \left[u_{UpXE}^2 + u_{UpYE}^2 (1 - e)^2 + u_{UpZE}^2 \right] \\ &= R'_S \left[u_{UpXE}^2 + u_{UpYE}^2 + u_{UpZE}^2 + u_{UpYE}^2 \left((1 - e)^2 - 1 \right) \right] \\ &= R'_S \left[1 + u_{UpYE}^2 \left((1 - e)^2 - 1 \right) \right] \end{aligned}$$

or with (5.1-10) of Section 5.1 squared:

$$\underline{u}_{Up} \cdot \underline{R}_S = R_0^2 / R_S'$$

Substituting in (5.2.1-2) obtains the relationship between R and R_S:

$$R^2 = R_S^2 + 2\,h\,R_0^2 / R_S' + h^2 \qquad (5.2.1\text{-}5)$$

5.2.2 POLAR COORDINATE ANGLE PARAMETERS

The cosine and sine of ϕ are parameters that will be useful later on, and are derived from Figure 5.2-1 as follows. First we substitute (5.1-9) into (5.2.1-1) to obtain in the E Frame:

$$\begin{aligned} R_{XE} &= u_{Up_{XE}} \left(R_S' + h \right) \\ R_{YE} &= u_{Up_{YE}} \left[(1 - e)^2 R_S' + h \right] \\ R_{ZE} &= u_{Up_{ZE}} \left(R_S' + h \right) \end{aligned} \qquad (5.2.2\text{-}1)$$

From Figure 5.2-1 we then write:

$$\cos\phi = \frac{R_{YE}}{R} \qquad \sin\phi = \frac{R_{Eq}}{R} = \frac{\sqrt{R_{XE}^2 + R_{ZE}^2}}{R} \qquad (5.2.2\text{-}2)$$

where

R_{Eq} = Projection of $\underline{R}$ on the equatorial plane.

Using (5.2.2-1), the cos ϕ term in (5.2.2-2) becomes:

$$\cos\phi = u_{Up_{YE}} \left[(1 - e)^2 R_S' + h \right] / R \qquad (5.2.2\text{-}3)$$

The term under the radical in (5.2.2-2) is with (5.2.2-1):

$$R_{XE}^2 + R_{ZE}^2 = \left(u_{Up_{XE}}^2 + u_{Up_{ZE}}^2 \right) \left(R_S' + h \right)^2 = \left(1 - u_{Up_{YE}}^2 \right) \left(R_S' + h \right)^2$$

Hence:

$$\sin\phi = \sqrt{1 - u_{Up_{YE}}^2} \left(R_S' + h \right) / R \qquad (5.2.2\text{-}4)$$

or equivalently:

$$\left(\frac{\sin \phi}{\sqrt{1 - u_{Up_{YE}}^2}}\right) = \left(R_S' + h\right) / R \tag{5.2.2-5}$$

The $\left(\frac{\sin \phi}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$ form in Equation (5.2.2-5) will prove more convenient for later use in handling singular cases near $u_{ZN_{YE}}$ equal to $\pm$ 1. This approach will also be used for some of the other parameters developed in this chapter.

5.2.3 LATITUDE ANGLE PARAMETERS

An expression for geodetic latitude (l) in Figure 5.2-1 can be obtained by inspection. The sine of l is equal to the projection of the unit vector $\underline{u}_{Up}$ on the earth polar axis:

$$\sin l = u_{Up_{YE}} \tag{5.2.3-1}$$

It will also be useful to determine expressions for the sine and cosine of ∂l in Figure 5.2-1, the difference between l and l_ϕ, which is also the angle between $\underline{R}$ and $\underline{u}_{Up}$. To derive these expressions, we start with Figure 5.2-1 for ∂l and some basic trigonometric identities:

$$\begin{aligned} \partial l &= l - l_\phi \\ \sin \partial l &= \sin l \cos l_\phi - \cos l \sin l_\phi \\ \cos \partial l &= \cos l \cos l_\phi + \sin l \sin l_\phi \end{aligned} \tag{5.2.3-2}$$

From Figure 5.2-1, l_ϕ is related to ϕ through:

$$l_\phi = \pi / 2 - \phi \tag{5.2.3-3}$$

hence,

$$\sin l_\phi = \cos \phi \qquad \cos l_\phi = \sin \phi \tag{5.2.3-4}$$

Substituting (5.2.3-1) with (5.2.3-4) into (5.2.3-2) obtains:

$$\begin{aligned} \sin \partial l &= u_{Up_{YE}} \sin \phi - \sqrt{1 - u_{Up_{YE}}^2} \cos \phi \\ \cos \partial l &= \sqrt{1 - u_{Up_{YE}}^2} \sin \phi + u_{Up_{YE}} \cos \phi \end{aligned}$$

We then substitute for sin ϕ and cos ϕ from (5.2.2-3) and (5.2.2-4):

$$\begin{aligned}\sin \partial l &= u_{Up\,YE} \sqrt{1 - u_{Up\,YE}^2}\, \left(R_S' + h\right) / R - \sqrt{1 - u_{Up\,YE}^2}\, u_{Up\,YE} \left[(1 - e)^2 R_S' + h\right] / R \\ &= u_{Up\,YE} \sqrt{1 - u_{Up\,YE}^2}\, \left[1 - (1 - e)^2\right] R_S' / R\end{aligned}$$

$$\begin{aligned}\cos \partial l &= \left(1 - u_{Up\,YE}^2\right)\left(R_S' + h\right) / R + u_{Up\,YE}^2 \left[(1 - e)^2 R_S' + h\right] / R \\ &= \left[\left(1 - u_{Up\,YE}^2 + u_{Up\,YE}^2 (1 - e)^2\right) R_S' + h\right] / R\end{aligned}$$

or, after rearrangement:

$$\begin{aligned}\left(\frac{\sin \partial l}{\sqrt{1 - u_{Up\,YE}^2}}\right) &= u_{Up\,YE} \left[1 - (1 - e)^2\right] R_S' / R \\ \cos \partial l &= \left\{\left(1 - u_{Up\,YE}^2 \left[1 - (1 - e)^2\right]\right) R_S' + h\right\} / R\end{aligned} \tag{5.2.3-5}$$

The Equation (5.2.3-5) $\left(\dfrac{\sin \partial l}{\sqrt{1 - u_{Up\,YE}^2}}\right)$ form will prove more convenient for later use in handling singular cases near $u_{ZN\,YE}$ equal to ± 1.

5.2.4 RADII OF CURVATURE

Consider the (4.4.1.1-5) transport rate equation using $\underline{\omega}_{EN}^E$ as the transport rate $\underline{\rho}^E$ (for more specificity):

$$\dot{\underline{u}}_{Up}^E = \underline{\omega}_{EN}^E \times \underline{u}_{Up}^E \tag{5.2.4-1}$$

where

$\underline{\omega}_{EN}^E$ = Angular rate of the N Frame relative to the E Frame in E Frame axes.

Taking the cross-product of (5.2.4-1) with $\underline{u}_{Up}^E$ and applying (3.1.1-16) provides the equivalent alternate form:

$$\underline{u}_{Up}^E \times \dot{\underline{u}}_{Up}^E = \underline{u}_{Up}^E \times \left(\underline{\omega}_{EN}^E \times \underline{u}_{Up}^E\right) = \underline{\omega}_{EN}^E - \underline{u}_{Up}^E \left(\underline{\omega}_{EN}^E \cdot \underline{u}_{Up}^E\right) \tag{5.2.4-2}$$

Identifying the term on the right as the horizontal component of $\underline{\omega}_{EN}^{E}$ (i.e., $\underline{\omega}_{EN}^{E}$ with the vertical component removed) then yields:

$$\underline{\omega}_{EN_H}^{E} = \underline{u}_{Up}^{E} \times \dot{\underline{u}}_{Up}^{E} \tag{5.2.4-3}$$

where

$\underline{\omega}_{EN_H}^{E}$ = Horizontal component of $\underline{\omega}_{EN}^{E}$.

Over an infinitesimal time interval dt, $\underline{\omega}_{EN_H}^{E}$ rotates the N Frame relative to the E Frame through an infinitesimal angle, and $\underline{u}_{Up}^{E}$ changes accordingly such that:

$$\underline{\omega}_{EN_H}^{E} = \frac{d\underline{\vartheta}_{EN_H}^{E}}{dt} \qquad \dot{\underline{u}}_{Up}^{E} = \frac{d\underline{u}_{Up}^{E}}{dt} \tag{5.2.4-4}$$

where

$d\underline{\vartheta}_{EN_H}^{E}$ = Infinitesimal angular rotation vector of the N relative to the E Frame over time interval dt.

$d\underline{u}_{Up}^{E}$ = Infinitesimal change in $\underline{u}_{Up}^{E}$ over time interval dt caused by $d\underline{\vartheta}_{EN_H}^{E}$.

Substituting (5.2.4-4) in (5.2.4-3) provides the equivalent differential form:

$$d\underline{\vartheta}_{EN_H}^{E} = \underline{u}_{Up}^{E} \times d\underline{u}_{Up}^{E} \tag{5.2.4-5}$$

We will now use (5.2.4-5) to find the radii of curvature associated with $d\underline{\vartheta}_{EN_H}^{E}$ and the corresponding horizontal position movement at the earth's surface and at the navigation system altitude. The derivation begins with the vector form of Equation (5.1-8) multiplied by $R_S^{\prime}$:

$$R_S^{\prime}\, \underline{u}_{Up}^{E} = \underline{R}_S^{E} + \left(\frac{1}{(1 - e)^2} - 1\right)\left(\underline{u}_{YE}^{E} \cdot \underline{R}_S^{E}\right) \underline{u}_{YE}^{E} \tag{5.2.4-6}$$

where

$\underline{R}_S^{E}$ = Position vector from earth's center to the navigation system earth surface location (See Figure 5.2-1) in E Frame coordinates.

$\underline{u}_{YE}^{E}$ = Unit vector along the E Frame Y axis in E Frame coordinates.

Taking the differential of (5.2.4-6) while recognizing $\underline{u}_{YE}^{E}$ as constant obtains:

$$R_S' \, d\underline{u}_{Up}^{E} + \underline{u}_{Up}^{E} \, dR_S' = d\underline{R}_S^{E} + \left(\frac{1}{(1 - e)^2} - 1 \right) \left(\underline{u}_{YE}^{E} \cdot d\underline{R}_S^{E} \right) \underline{u}_{YE}^{E}$$

or upon rearrangement:

$$R_S' \, d\underline{u}_{Up}^{E} = d\underline{R}_S^{E} + \left(\frac{1}{(1 - e)^2} - 1 \right) \left(\underline{u}_{YE}^{E} \cdot d\underline{R}_S^{E} \right) \underline{u}_{YE}^{E} - dR_S' \, \underline{u}_{Up}^{E} \tag{5.2.4-7}$$

where

d( ) = Differential change in ( ) over time interval dt.

Multiplying (5.2.4-5) by R_S' and substituting (5.2.4-7) for $R_S' \, d\underline{u}_{Up}^{E}$ then finds:

$$R_S' \, d\underline{\vartheta}_{EN_H}^{E} = \underline{u}_{Up}^{E} \times d\underline{R}_S^{E} + \left(\frac{1}{(1 - e)^2} - 1 \right) \left(\underline{u}_{Up}^{E} \times \underline{u}_{YE}^{E} \right) \left(\underline{u}_{YE}^{E} \cdot d\underline{R}_S^{E} \right) \tag{5.2.4-8}$$

Equation (5.2.4-8) equates $d\underline{\vartheta}_{EN_H}^{E}$ to the associated $d\underline{R}_S^{E}$ earth surface position movement. Due to the symmetry of the earth surface shape ellipsoid about the earth's polar axis, it will prove expeditious to find north and east components of (5.2.4-8) in our quest for radii of curvature. Toward this end we decompose $\underline{u}_{YE}^{E}$ in (5.2.4-8) into its north and vertical components (the east component is zero, i.e., the component of a unit vector along the E Frame Y axis in Figure 5.1-1 that is normal to the page):

$$\underline{u}_{YE}^{E} = \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E} \right) \underline{u}_{North}^{E} + \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{Up}^{E} \right) \underline{u}_{Up}^{E} \tag{5.2.4-9}$$

where

$\underline{u}_{North}^{E}$ = Unit vector in the horizontal north direction in E Frame coordinates.

Using (5.2.4-9) and recognizing that the cross-product of $\underline{u}_{Up}^{E}$ with $\underline{u}_{North}^{E}$ lies west (i.e., negative east), the $\underline{u}_{Up}^{E} \times \underline{u}_{YE}^{E}$ term in (5.2.4-8) becomes:

$$\underline{u}_{Up}^{E} \times \underline{u}_{YE}^{E} = \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E} \right) \underline{u}_{Up}^{E} \times \underline{u}_{North}^{E} = - \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E} \right) \underline{u}_{East}^{E} \tag{5.2.4-10}$$

where

$\underline{u}_{East}^{E}$ = Unit vector in the horizontal east direction in E Frame coordinates.

With (5.2.4-9), the $\underline{u}_{YE}^{E} \cdot d\underline{R}_S^{E}$ term in (5.2.4-8) is:

$$\underline{u}_{YE}^{E} \cdot d\underline{R}_{S}^{E} = \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E}\right)\left(\underline{u}_{North}^{E} \cdot d\underline{R}_{S}^{E}\right) + \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{Up}^{E}\right)\left(\underline{u}_{Up}^{E} \cdot d\underline{R}_{S}^{E}\right) \tag{5.2.4-11}$$

We will now formally show that the $\underline{u}_{Up}^{E} \cdot d\underline{R}_{S}^{E}$ term in (5.2.4-11) is zero using Equation (5.2.1-1) rearranged in E Frame coordinates:

$$\underline{R}_{S}^{E} = \underline{R}^{E} - h\,\underline{u}_{Up}^{E} \tag{5.2.4-12}$$

The differential of (5.2.4-12) is:

$$d\underline{R}_{S}^{E} = d\underline{R}^{E} - \underline{u}_{Up}^{E}\,dh - h\,d\underline{u}_{Up}^{E} = d\underline{R}_{H}^{E} - h\,d\underline{u}_{Up}^{E} \tag{5.2.4-13}$$

where

$d\underline{R}_{H}^{E}$ = Horizontal component of $d\underline{R}^{E}$ (i.e., $d\underline{R}^{E}$ with the vertical dh component removed).

Taking the dot product of (5.2.4-13) with $\underline{u}_{Up}^{E}$:

$$\underline{u}_{Up}^{E} \cdot d\underline{R}_{S}^{E} = \underline{u}_{Up}^{E} \cdot d\underline{R}_{H}^{E} - h\,\underline{u}_{Up}^{E} \cdot d\underline{u}_{Up}^{E} = -\,h\,\underline{u}_{Up}^{E} \cdot d\underline{u}_{Up}^{E} \tag{5.2.4-14}$$

But, $\underline{u}_{Up}^{E} \cdot \underline{u}_{Up}^{E} = 1$, hence, from the differential, $\underline{u}_{Up}^{E} \cdot d\underline{u}_{Up}^{E} = 0$. Therefore, (5.2.4-14) shows that $\underline{u}_{Up}^{E} \cdot d\underline{R}_{S}^{E}$ is zero and (5.2.4-11) reduces to:

$$\underline{u}_{YE}^{E} \cdot d\underline{R}_{S}^{E} = \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E}\right)\left(\underline{u}_{North}^{E} \cdot d\underline{R}_{S}^{E}\right) \tag{5.2.4-15}$$

We now substitute (5.2.4-10) and (5.2.4-15) into (5.2.4-8) and apply generalized Equation (3.1.1-10) to obtain:

$$R_{S}'\,d\underline{\vartheta}_{EN_H}^{E} = \left[\left(\underline{u}_{Up}^{E}\times\right) - \left(\frac{1}{(1-e)^2} - 1\right)\left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E}\right)^2 \underline{u}_{East}^{E}\left(\underline{u}_{North}^{E}\right)^T\right] d\underline{R}_{S}^{E} \tag{5.2.4-16}$$

As a final step, the $\left(\dfrac{1}{(1-e)^2} - 1\right)\left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E}\right)^2$ term in (5.2.4-16) can be expressed in terms of R_S', R_0 and e by first taking the dot product of (5.2.4-9) with itself and equating the result to 1 (the magnitude of $\underline{u}_{YE}^{E}$):

$$\left(\underline{u}_{YE}^{E} \cdot \underline{u}_{North}^{E}\right)^2 + \left(\underline{u}_{YE}^{E} \cdot \underline{u}_{Up}^{E}\right)^2 = 1$$

or

$$\left(\underline{u}^E_{YE} \cdot \underline{u}^E_{North}\right)^2 = 1 - \left(\underline{u}^E_{YE} \cdot \underline{u}^E_{Up}\right)^2 = 1 - u^2_{Up_{YE}}$$

Thus,

$$\begin{aligned}
\left(\frac{1}{(1-e)^2} - 1\right)\left(\underline{u}^E_{YE} \cdot \underline{u}^E_{North}\right)^2 &= \left(\frac{1}{(1-e)^2} - 1\right)\left(1 - u^2_{Up_{YE}}\right) \\
&= \frac{1}{(1-e)^2}\left(1 - u^2_{Up_{YE}}\right) + u^2_{Up_{YE}} - 1 \\
&= \frac{1}{(1-e)^2}\left[(1-e)^2 u^2_{Up_{YE}} + 1 - u^2_{Up_{YE}}\right] - 1 \\
&= \frac{1}{(1-e)^2}\left[1 + u^2_{Up_{YE}}\left((1-e)^2 - 1\right)\right] - 1
\end{aligned} \tag{5.2.4-17}$$

Comparing the form of Equation (5.2.4-17) with the reciprocal of Equation (5.1-10) squared shows that:

$$\left(\frac{1}{(1-e)^2} - 1\right)\left(\underline{u}^E_{YE} \cdot \underline{u}^E_{North}\right)^2 = \frac{1}{(1-e)^2}\frac{R_0^2}{R_S'^2} - 1 \tag{5.2.4-18}$$

with which Equation (5.2.4-16) becomes:

$$R_S'\, d\underline{\vartheta}^E_{EN_H} = \left[\left(\underline{u}^E_{Up}\times\right) - \left(\frac{1}{(1-e)^2}\frac{R_0^2}{R_S'^2} - 1\right)\underline{u}^E_{East}\left(\underline{u}^E_{North}\right)^T\right] d\underline{R}^E_S \tag{5.2.4-19}$$

Equation (5.2.4-19) is now in a form that can be used to define radii of curvature associated with $d\underline{\vartheta}^E_{EN_H}$ and the corresponding horizontal earth surface position movement $d\underline{R}^E_S$. The radii of curvature will be defined in terms of north and east components of linear and angular movement, so we first transform to the geographic Geo Frame for analysis. Multiplying (5.2.4-19) by C^{Geo}_E and expanding:

$$\begin{aligned}
R_S'\, C^{Geo}_E\, d\underline{\vartheta}^E_{EN_H} &= \left[C^{Geo}_E\left(\underline{u}^E_{Up}\times\right) - \left(\frac{1}{(1-e)^2}\frac{R_0^2}{R_S'^2} - 1\right)C^{Geo}_E\,\underline{u}^E_{East}\left(\underline{u}^E_{North}\right)^T\right]\left(C^{Geo}_E\right)^T C^{Geo}_E\, d\underline{R}^E_S \\
&= \left[C^{Geo}_E\left(\underline{u}^E_{Up}\times\right)\left(C^{Geo}_E\right)^T - \left(\frac{1}{(1-e)^2}\frac{R_0^2}{R_S'^2} - 1\right)\underline{u}^{Geo}_{East}\left(\underline{u}^E_{North}\right)^T\left(C^{Geo}_E\right)^T\right] d\underline{R}^{Geo}_{SE}
\end{aligned} \tag{5.2.4-20}$$

or, after applying (3.1.1-25) and (3.1.1-39):

$$R'_S \, d\underline{\vartheta}^{Geo}_{EN_H} = \left[\left(\underline{u}^{Geo}_{Up} \times \right) - \left(\frac{1}{(1 - e)^2} \frac{R_0^2}{R'^2_S} - 1 \right) \underline{u}^{Geo}_{East} \left(\underline{u}^{Geo}_{North} \right)^T \right] d\underline{R}^{Geo}_{SE} \tag{5.2.4-21}$$

where

$d\underline{R}^{Geo}_{SE}$ = $d\underline{R}^E_S$ transformed to the Geo Frame. The SE subscript notation has been adopted to specifically identify that the surface position movement is relative to the E Frame even though the components are being projected on Geo Frame coordinates.

The north component of (5.2.4-21) is found by taking the dot product with $\underline{u}^{Geo}_{North}$, applying (3.1.1-35), and recognizing that the cross-product of north with up unit vectors lies east:

$$\begin{aligned} R'_S \, d\vartheta_{EN_{North}} &= \underline{u}^{Geo}_{North} \cdot \left(\underline{u}^{Geo}_{Up} \times d\underline{R}^{Geo}_{SE} \right) = d\underline{R}^{Geo}_{SE} \cdot \left(\underline{u}^{Geo}_{North} \times \underline{u}^{Geo}_{Up} \right) \\ &= d\underline{R}^{Geo}_{SE} \cdot \underline{u}^{Geo}_{East} = dR_{SE_{East}} \end{aligned}$$

or

$$d\vartheta_{EN_{North}} = \frac{1}{R'_S} \, dR_{SE_{East}} \tag{5.2.4-22}$$

where

$d\vartheta_{EN_{North}}, dR_{SE_{East}}$ = The north component of $d\underline{\vartheta}_{EN_H}$ and the east component of $d\underline{R}_{SE}$.

The east component of (5.2.4-21) is found by taking the dot product with $\underline{u}^{Geo}_{East}$, applying (3.1.1-25) and (3.1.1-35), while recognizing that the cross-product of east with up unit vectors lies south (i.e., minus north):

$$\begin{aligned} R'_S \, d\vartheta_{EN_{East}} &= \underline{u}^{Geo}_{East} \cdot \left(\underline{u}^{Geo}_{Up} \times d\underline{R}^{Geo}_{SE} \right) - \left(\frac{1}{(1 - e)^2} \frac{R_0^2}{R'^2_S} - 1 \right) dR_{SE_{North}} \\ &= d\underline{R}^{Geo}_{SE} \cdot \left(\underline{u}^{Geo}_{East} \times \underline{u}^{Geo}_{Up} \right) - \left(\frac{1}{(1 - e)^2} \frac{R_0^2}{R'^2_S} - 1 \right) dR_{SE_{North}} \\ &= - \, d\underline{R}^{Geo}_{SE} \cdot \underline{u}^{Geo}_{North} - \left(\frac{1}{(1 - e)^2} \frac{R_0^2}{R'^2_S} - 1 \right) dR_{SE_{North}} \end{aligned}$$

(Continued)

(Continued)

$$= -\frac{1}{(1-e)^2}\frac{R_0^2}{R_S'^{\,2}}\, dR_{SE_{North}}$$

or

$$d\vartheta_{EN_{East}} = -\frac{1}{(1-e)^2 \dfrac{R_S'^{\,3}}{R_0^2}}\, dR_{SE_{North}} \tag{5.2.4-23}$$

where

$d\vartheta_{EN_{East}}, dR_{SE_{North}}$ = The east component of $d\underline{\vartheta}_{EN_H}$ and the north component of $d\underline{R}_{SE}$.

Radii of curvature corresponding to horizontal north and east earth surface position movements are defined as the ratio of the horizontal earth surface position movement divided by the corresponding horizontal angular rotation over the earth's surface. From Equations (5.2.4-22) and (5.2.4-23) we see then that the radii of curvature are given by:

$$r_{Ls} \equiv \frac{dR_{SE_{East}}}{d\vartheta_{EN_{North}}} = R_S' \tag{5.2.4-24}$$

$$r_{ls} \equiv \frac{dR_{SE_{North}}}{-\,d\vartheta_{EN_{East}}} = (1-e)^2\frac{R_S'^{\,3}}{R_0^2} \tag{5.2.4-25}$$

where

r_{Ls}, r_{ls} = Radii of curvature at the earth surface corresponding to longitude (East), latitude (North) movements of $\underline{R}_S$.

With the (5.2.4-24) - (5.2.4-25) definitions, Equations (5.2.4-22) and (5.2.4-23) become:

$$d\vartheta_{EN_{North}} = \frac{1}{r_{Ls}}\, dR_{SE_{East}} \qquad d\vartheta_{EN_{East}} = -\frac{1}{r_{ls}}\, dR_{SE_{North}} \tag{5.2.4-26}$$

The vector form of (5.2.4-26) is found from the following development:

$$d\underline{\vartheta}_{EN_H}^{Geo} = \begin{bmatrix} 0 & -\frac{1}{r_{ls}} & 0 \\ \frac{1}{r_{Ls}} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} d\underline{R}_{SE}^{Geo} = \begin{bmatrix} \frac{1}{r_{ls}} & 0 & 0 \\ 0 & \frac{1}{r_{Ls}} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} d\underline{R}_{SE}^{Geo}$$

which is equivalently:

$$d\underline{\vartheta}_{EN_H}^{Geo} = F_{CS}^{Geo}\left(\underline{u}_{Up}^{Geo} \times d\underline{R}_{SE}^{Geo}\right) \qquad (5.2.4\text{-}27)$$

in which

$$F_{CS}^{Geo} \equiv \begin{bmatrix} \frac{1}{r_{ls}} & 0 & 0 \\ 0 & \frac{1}{r_{Ls}} & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad (5.2.4\text{-}28)$$

where

F_{CS}^{Geo} = Geo Frame version of the curvature matrix associated with earth surface referenced horizontal motion.

The Equation (5.2.4-27) and (5.2.4-28) forms will now be used to find radii of curvature corresponding to horizontal position motion at the navigation system altitude (which may be above or below the earth surface). We begin with the Geo Frame version of (5.2.4-13):

$$d\underline{R}_{SE}^{Geo} = d\underline{R}_{HE}^{Geo} - h\, d\underline{u}_{UpE}^{Geo} \qquad (5.2.4\text{-}29)$$

where

$d\underline{R}_{HE}^{Geo}$, $d\underline{u}_{UpE}^{Geo}$ = $d\underline{R}_{H}^{E}$ and $d\underline{u}_{Up}^{E}$ transformed to the Geo Frame. The added E in the subscripts identifies the differential change to be relative to the E Frame.

Substituting (5.2.4-29) in the (5.2.4-27) cross-product term finds:

$$\underline{u}_{Up}^{Geo} \times d\underline{R}_{SE}^{Geo} = \underline{u}_{Up}^{Geo} \times d\underline{R}_{HE}^{Geo} - h\, \underline{u}_{Up}^{Geo} \times d\underline{u}_{UpE}^{Geo} \qquad (5.2.4\text{-}30)$$

But from the Geo Frame version of (5.2.4-5):

$$\underline{u}_{Up}^{Geo} \times d\underline{u}_{UpE}^{Geo} = d\underline{\vartheta}_{EN_H}^{Geo} \qquad (5.2.4\text{-}31)$$

Substituting (5.2.4-31) in (5.2.4-30) and applying the result to (5.2.4-27) then shows (with rearrangement) that:

$$\left(I + h\, F_{CS}^{Geo}\right) d\underline{\vartheta}_{EN_H}^{Geo} = F_{CS}^{Geo}\left(\underline{u}_{Up}^{Geo} \times d\underline{R}_{HE}^{Geo}\right) \qquad (5.2.4\text{-}32)$$

or

$$d\underline{\vartheta}_{EN_H}^{Geo} = F_C^{Geo}\left(\underline{u}_{Up}^{Geo} \times d\underline{R}_{HE}^{Geo}\right) \tag{5.2.4-33}$$

with

$$F_C^{Geo} \equiv \left(I + h\, F_{CS}^{Geo}\right)^{-1} F_{CS}^{Geo} \tag{5.2.4-34}$$

where

F_C^{Geo} = Geo Frame version of the curvature matrix associated with horizontal motion at altitude.

Using (5.2.4-28) for F_{CS}^{Geo}, the F_C^{Geo} curvature matrix is evaluated in steps as follows:

$$I + h\, F_{CS}^{Geo} = \begin{bmatrix} 1 + \dfrac{h}{r_{l_S}} & 0 & 0 \\ 0 & 1 + \dfrac{h}{r_{L_S}} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \dfrac{r_{l_S} + h}{r_{l_S}} & 0 & 0 \\ 0 & \dfrac{r_{L_S} + h}{r_{L_S}} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

from which

$$F_C^{Geo} = \left(I + h\, F_{CS}^{Geo}\right)^{-1} F_{CS}^{Geo} = \begin{bmatrix} \dfrac{1}{r_{l_S} + h} & 0 & 0 \\ 0 & \dfrac{1}{r_{L_S} + h} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5.2.4-35}$$

Comparing (5.2.4-35) with (5.2.4-28), we see that the equivalent radii of curvature at altitude are given by:

$$r_L = r_{L_S} + h \tag{5.2.4-36}$$

$$r_l = r_{l_S} + h \tag{5.2.4-37}$$

where

r_L, r_l = Equivalent radii of curvature at the actual (at altitude) position location corresponding to longitude (East), latitude (North) movements of $\underline{R}$ and $\underline{u}_{Up}$ (See Figure 5.2-1).

With (5.2.4-36) and (5.2.4-37), Equation (5.2.4-35) for the curvature matrix at altitude becomes:

$$F_C^{Geo} = \begin{bmatrix} \frac{1}{r_l} & 0 & 0 \\ 0 & \frac{1}{r_L} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5.2.4-38}$$

5.3 TRANSPORT RATE

Transport rate $\underline{\rho}$ is defined in Section 4.1.1 as the angular rate of the locally level N Frame relative to the E Frame (see Section 2.2 for N Frame definition). The horizontal components of $\underline{\rho}$ (i.e., the N Frame X, Y components) rotate the N Frame Z axis to remain parallel to the geodetic vertical (i.e., the $\underline{u}_{Up}$ vector). As discussed in Section 4.5, the N Frame vertical axis (Z) component of $\underline{\rho}$ is somewhat arbitrary, and depends on the type of N Frame implementation selected.

From the previous description of $\underline{\rho}$, we see that horizontal transport rate is a function of the $\underline{u}_{Up}$ angular rate (which is produced by horizontal velocity over the earth). To derive the equations relating horizontal velocity to horizontal transport rate, we return to Equation (5.2.4-33) and divide it by the infinitesimal time interval dt:

$$\frac{d\underline{\vartheta}_{EN_H}^{Geo}}{dt} = F_C^{Geo}\left(\underline{u}_{Up}^{Geo} \times \frac{d\underline{R}_{HE}^{Geo}}{dt}\right) \tag{5.3-1}$$

The definition for $d\underline{R}_{HE}^{Geo}$ given in Section 5.2.4 (following Equation (5.2.4-29)) is analytically:

$$d\underline{R}_{HE}^{Geo} = C_E^{Geo}\, d\underline{R}_H^E \tag{5.3-2}$$

Based on (5.3-2) and the definition for velocity relative to the earth ($\underline{v}$) in (4.3-1), we see that

$$\frac{d\underline{R}_{HE}^{Geo}}{dt} = \underline{v}_H^{Geo} \tag{5.3-3}$$

where

$\underline{v}_H^{Geo}$ = Horizontal component of velocity relative to the earth in the Geo Frame.

We also know from the Geo Frame form of the first equation in (5.2.4-4) and the transport rate definition that:

$$\underline{\rho}_H^{Geo} \equiv \underline{\omega}_{EN_H}^{Geo} = \frac{d\underline{\vartheta}_{EN_H}^{Geo}}{dt} \tag{5.3-4}$$

where

$\underline{\rho}_H^{Geo}$ = Horizontal components of transport rate in the Geo Frame.

Thus, with (5.3-3) and (5.3-4), Equation (5.3-1) becomes the horizontal transport rate equation in the Geo Frame:

$$\underline{\rho}_H^{Geo} = F_C^{Geo}\left(\underline{u}_{Up}^{Geo} \times \underline{v}^{Geo}\right) \tag{5.3-5}$$

Note that we have used the total velocity $\underline{v}^{Geo}$ in (5.3-5) (rather than $\underline{v}_H^{Geo}$) because the $\underline{u}_{Up}^{Geo}$ cross-product cancels the vertical $\underline{v}^{Geo}$ component.

The components of horizontal transport rate in the N Frame are obtained by multiplying (5.3-5) by the C_{Geo}^N transformation matrix and expanding using (3.2.1-4):

$$C_{Geo}^N \underline{\rho}_H^{Geo} = C_{Geo}^N F_C^{Geo}\left(C_{Geo}^N\right)^T C_{Geo}^N\left(\underline{u}_{Up}^{Geo} \times \underline{v}^{Geo}\right)$$

or after applying generalized Equation (3.1.1-42):

$$\underline{\rho}_H^N = F_C^N\left(\underline{u}_{Up}^N \times \underline{v}^N\right) \tag{5.3-6}$$

with

$$F_C^N \equiv C_{Geo}^N F_C^{Geo}\left(C_{Geo}^N\right)^T \tag{5.3-7}$$

where

F_C^N = N Frame version of the curvature matrix associated with horizontal motion at altitude.

The N Frame is rotated from the Geo Frame about the local vertical (Z axis of the Geo and N Frames) by the wander angle shown in Figure 4.4.2.1-1. Taking the transpose of Equation (4.3.1-1) gives for the C_{Geo}^N matrix:

$$C_{Geo}^N = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.3-8}$$

where

α = Wander angle.

Using (5.3-8) for C_{Geo}^{N} in (5.3-7) with (5.2.4-38) for F_{C}^{Geo}, we see that:

$$F_C^N = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{r_l} & 0 & 0 \\ 0 & \frac{1}{r_L} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{r_l}\cos^2\alpha + \frac{1}{r_L}\sin^2\alpha & \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\sin\alpha\cos\alpha & 0 \\ \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\sin\alpha\cos\alpha & \frac{1}{r_l}\sin^2\alpha + \frac{1}{r_L}\cos^2\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5.3-9}$$

But,

$$\begin{aligned} \frac{1}{r_l}\cos^2\alpha + \frac{1}{r_L}\sin^2\alpha &= \frac{1}{r_l} + \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\sin^2\alpha \\ \frac{1}{r_l}\sin^2\alpha + \frac{1}{r_L}\cos^2\alpha &= \frac{1}{r_l} + \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\cos^2\alpha \end{aligned} \tag{5.3-10}$$

Substituting (5.3-10) in (5.3-9) then finds:

$$F_C^N = \begin{bmatrix} \frac{1}{r_l} + \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\sin^2\alpha & \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\sin\alpha\cos\alpha & 0 \\ \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\sin\alpha\cos\alpha & \frac{1}{r_l} + \left(\frac{1}{r_L} - \frac{1}{r_l}\right)\cos^2\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5.3-11}$$

Equations (4.4.2.1-2) show that the D_{21} and D_{22} elements of the C_N^E direction cosine matrix are equal, respectively, to the product of cosine latitude (l) with sin α and cos α. Then, with Equation (5.2.3-1), the sin α, cos α terms in (5.3-11) are:

$$\sin\alpha = \frac{D_{21}}{\cos l} = \frac{D_{21}}{\sqrt{1 - u_{Up_{YE}}^2}} \qquad \cos\alpha = \frac{D_{22}}{\cos l} = \frac{D_{22}}{\sqrt{1 - u_{Up_{YE}}^2}} \tag{5.3-12}$$

The bracketed reciprocal radius of curvature difference term in (5.3-11) can be rearranged using (5.2.4-36) and (5.2.4-37) as follows:

$$\left(\frac{1}{r_L} - \frac{1}{r_l}\right) = \frac{r_l - r_L}{r_L\, r_l} = \frac{r_{ls} - r_{Ls}}{r_L\, r_l} = \frac{1}{r_l}\,\frac{r_{Ls}}{r_L}\left(\frac{r_{ls}}{r_{Ls}} - 1\right) \tag{5.3-13}$$

With (5.2.4-36) and (5.2.4-24), the $\frac{r_{Ls}}{r_L}$ term in (5.3-13) is given by:

$$\frac{r_{Ls}}{r_L} = \frac{r_{Ls}}{r_{Ls} + h} = \frac{1}{1 + h / R_S'} \tag{5.3-14}$$

Combining (5.2.4-24), (5.2.4-25) and (5.1-10) squared, the $\frac{r_{ls}}{r_{Ls}}$ -1 term in (5.3-13) is:

$$\begin{aligned}\frac{r_{ls}}{r_{Ls}} - 1 &= (1 - e)^2 \frac{R_S'^{\,2}}{R_0^2} - 1 \\ &= \frac{(1 - e)^2}{1 + u_{Up_{YE}}^2\left[(1 - e)^2 - 1\right]} - 1 = \frac{\left(1 - u_{Up_{YE}}^2\right)\left[(1 - e)^2 - 1\right]}{1 + u_{Up_{YE}}^2\left[(1 - e)^2 - 1\right]}\end{aligned} \tag{5.3-15}$$

Equations (4.4.2.1-2) show that the D_{23} element of the C_N^E direction cosine matrix is sin *l*, hence, from (5.2.3-1):

$$u_{Up_{YE}} = D_{23} \tag{5.3-16}$$

Substituting (5.3-14) and (5.3-15) into (5.3-13), the result with (5.3-12) into (5.3-11), and using (5.3-16) for $u_{Up_{YE}}$ yields the desired expression for F_C^N, the N Frame curvature matrix at altitude. Substituting (5.3-6) (with $\underline{u}_{Up}^N$ equated to $\underline{u}_{ZN}^N$ by the N Frame Z axis definition) into (4.1.1-6) then obtains the total $\underline{\rho}^N$ N Frame transport rate equation based on F_C^N, the N Frame velocity $\underline{v}^N$, and the selected vertical component of $\underline{\rho}^N$ (from Section 4.5). The overall results are summarized as follows:

$$\underline{\rho}^N = F_C^N\left(\underline{u}_{ZN}^N \times \underline{v}^N\right) + \rho_{ZN}\, \underline{u}_{ZN}^N \tag{5.3-17}$$

with

$$\underline{u}_{ZN}^{N} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$$

$$F_C^N = \begin{bmatrix} F_{C_{11}} & F_{C_{12}} & 0 \\ F_{C_{21}} & F_{C_{22}} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$F_{C_{11}} = \frac{1}{r_l}\left(1 + D_{21}^2 \, f_{eh}\right) \qquad F_{C_{12}} = \frac{1}{r_l} D_{21} \, D_{22} \, f_{eh} \tag{5.3-18}$$

$$F_{C_{21}} = \frac{1}{r_l} D_{21} \, D_{22} \, f_{eh} \qquad F_{C_{22}} = \frac{1}{r_l}\left(1 + D_{22}^2 \, f_{eh}\right)$$

$$f_e = \frac{(1 - e)^2 - 1}{1 + D_{23}^2\left[(1 - e)^2 - 1\right]} \qquad f_h = \frac{1}{1 + h / R_S'} \qquad f_{eh} = f_e \, f_h$$

in which R_S' is given by (5.1-10) with (5.3-16) for $u_{Up_{YE}}$, r_l is provided by (5.2.4-37) with (5.2.4-25) for r_{ls}, and where:

$\underline{\rho}^N$ = Transport rate vector in N Frame coordinates (i.e., angular rate of the local level navigation N Frame relative to the earth fixed E Frame).

ρ_{ZN} = Vertical (Z axis) component of $\underline{\rho}^N$ (see Section 4.5 for options).

$\underline{v}^N$ = Velocity vector relative to the earth in N Frame axes.

$\underline{u}_{ZN}^N$ = Unit vector along the N Frame Z axis (i.e., upward).

F_C^N = Curvature matrix in the N Frame.

D_{2j} = Element j in the second row of the C_N^E matrix.

r_l = Local radius of curvature at altitude in North/South (latitude change) direction.

h = Altitude.

R_S' = Modified distance from earth center to the local earth surface referenced position location.

e = Earth's ellipticity (equals 1/298.3).

5.4 GRAVITY MODEL

The standard gravity model utilized in most inertial navigation systems is based on the generalized model given in Reference 3 - Section 4.4 and Reference 4 - Section 5. Reference 3 shows that for a positive altitude, the gravity components at the actual position location can be accurately approximated by:

For $h \geq 0$:

$$g_r = -\frac{\mu}{R^2}\left[1 - \frac{3}{2} J_2 \left(\frac{R_0}{R}\right)^2 \left(3 \cos^2 \phi - 1\right) - 2 J_3 \left(\frac{R_0}{R}\right)^3 \cos \phi \left(5 \cos^2 \phi - 3\right) - \cdots\right]$$

$$\left(\frac{g_\phi}{\sin \phi}\right) = 3 \frac{\mu}{R^2}\left(\frac{R_0}{R}\right)^2 \left[J_2 \cos \phi + \frac{1}{2} J_3 \frac{R_0}{R}\left(5 \cos^2 \phi - 1\right) + \cdots\right] \tag{5.4-1}$$

$$g_\theta \approx 0$$

where, with reference to Figures 5.1-1 and 5.2-1 given previously:

g_r = Component of gravity along the $\underline{R}$ direction.

g_ϕ = Component of gravity perpendicular to $\underline{R}$ in the local meridian plane (positive in the plus ϕ direction).

g_θ = Component of gravity perpendicular to $\underline{R}$ and perpendicular to the local meridian plane.

R = Magnitude of $\underline{R}$.

R_0 = Earth equatorial radius.

ϕ = Angle from earth's positive polar rotation axis to $\underline{R}$.

μ = Product of the mass of the earth with the universal gravitational constant. See Table 5.6-1 for numerical value.

$J_2, J_3, \cdots$ = Small empirical constants that are functions of the mass distribution of the earth. For a spherical earth of constant density, $J_2, J_3, \cdots$ would be zero. See Table 5.6-1 for numerical values.

The $(g_\phi / \sin \phi)$ format in (5.4-1) is used to avoid singularity problems in subsequent developments near $\phi = 0$ and $\phi = \pi$. The R^2 and $\cos \phi$ terms in (5.4-1) are provided by Equations (5.2.1-5) and (5.2.2-3), using (5.2.1-4) for R_S, (5.1-10) for R_S' and (5.3-16) for $u_{Up_{YE}}$.

For negative altitude h (e.g., below sea level or on the earth's surface in some landlocked regions such as Death Valley, Nevada, US), the author has no knowledge of any "official" gravity model in the classical navigation literature. For negative h locations, a reasonable approximation for gravity can be formulated based on the classical inverse square law gravity model within a uniform sphere of radius R and constant mass density. Consider a point within the sphere at distance r from the center. Further, consider the mass of the sphere as being divided into two parts; Part 1 being the mass at distance r or less from the center of the sphere, and Part 2 being the mass at distance greater that r from the center. Reference 37 - Chapter 5, Section 14 shows that the gravity potential produced by the Part 1 mass is the same as if the Part 1 mass was concentrated at the center of the sphere. Therefore, the gravitational acceleration produced by the Part 1 mass (i.e., the gradient of the potential) is the same as if the Part 1 mass was concentrated at the center. If we think of the Part 2 mass as consisting of a composite of thin spherical shells of increasing radius, Reference 37 - Chapter 5, Section 14 shows that the gravitational potential from each shell is zero at distance r from the center (i.e., within each shell). Hence, the gravitational force produced at point r by all the shells (i.e., the total Part 2 mass) is zero. Thus, the total gravitational acceleration at point r is that resulting from only the Part 1 mass which, using Newton's inverse square gravity law, is given by $g = G\, m / r^2 = G\, \rho \frac{4}{3} \pi\, r^3 / r^2 = G\, \rho \frac{4}{3} \pi\, r$ in which g is the gravitational acceleration at r, G is the universal gravitational constant, m is the Part 1 mass, and ρ is the mass density of the sphere. We see then, that in a constant density sphere, gravity increases linearly with distance from the center. For earth gravity when h is negative, we can assume that a similar approximately linear relationship exists, however, to give it accuracy, we also stipulate that the negative h gravity model will equal the Equation (5.4-1) positive h model for zero h. A gravity model that meets these criteria is given by:

For $h < 0$:

$$g_r = \frac{R}{R_S}\, g_{r_S} \qquad \left(\frac{g_\phi}{\sin\phi}\right) = \frac{R}{R_S}\left(\frac{g_\phi}{\sin\phi}\right)_S \qquad g_\theta \approx 0 \tag{5.4-2}$$

where

$g_{r_S}, \left(\dfrac{g_\phi}{\sin\phi}\right)_S$ = Values for g_r and $\left(\dfrac{g_\phi}{\sin\phi}\right)$ calculated using Equation (5.4-1) with R set to R_S. The R_S value is calculated using Equations (5.2.3-1), (5.1-10) and (5.2.1-4).

In most inertial navigation systems, gravity components are required along navigation coordinates that are aligned with the local horizontal and geodetic vertical, i.e., $\underline{u}_{Up}$ (see Figure 5.2-1 of Section 5.2). Components of gravity along $\underline{u}_{Up}$ and horizontal North ($\underline{u}_{North}$ in Figure 5.2-1) can be calculated by transforming the g_r, g_ϕ components through the ∂l angle (see Figure

5.2-1). Once the North and Up gravity components are calculated, components in the navigation N Frame (Section 2.2 definition) can be computed by transformation through the wander angle. The North, Up gravity components are calculated as:

$$g_{Up} = g_r \cos \partial l - g_\phi \sin \partial l \qquad g_{North} = - g_\phi \cos \partial l - g_r \sin \partial l \tag{5.4-3}$$

where

g_{Up} = Gravity component along $\underline{u}_{Up}$.

g_{North} = Gravity component along $\underline{u}_{North}$.

In terms of parameters previously derived in Section 5.2 and in this section, the previous expressions are equivalently:

$$\begin{aligned} g_{Up} &= g_r \cos \partial l - \left(\frac{g_\phi}{\sin \phi}\right)\left(\frac{\sin \phi}{\sqrt{1 - u_{Up_{YE}}^2}}\right)\left(\frac{\sin \partial l}{\sqrt{1 - u_{Up_{YE}}^2}}\right)\left(1 - u_{Up_{YE}}^2\right) \\ \left(\frac{g_{North}}{\sqrt{1 - u_{Up_{YE}}^2}}\right) &= - \left(\frac{g_\phi}{\sin \phi}\right)\left(\frac{\sin \phi}{\sqrt{1 - u_{Up_{YE}}^2}}\right) \cos \partial l - g_r \left(\frac{\sin \partial l}{\sqrt{1 - u_{Up_{YE}}^2}}\right) \end{aligned} \tag{5.4-4}$$

with the contributing terms provided by Equations (5.4-1), (5.4-2), (5.2.2-5), (5.2.3-5) and (5.3-16). The $\left(\frac{g_{North}}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$ format in (5.4-4) will prove useful in avoiding singularities near $u_{Up_{YE}} = \pm 1$ when calculating the navigation N Frame components of gravity.

5.4.1 PLUMB-BOB GRAVITY

For inertial navigation equations written in locally level geodetic vertical coordinates, gravity appears in conjunction with an earth rate based centripetal acceleration term. As described in Section 4.3, the sum of these two terms is called "plumb-bob" gravity because it acts along the line a plumb-bob would take when stationary relative to the earth at the same position location. The vector expression for plumb-bob gravity is given by (4.3-15) repeated below without coordinate frame designation:

$$\underline{g}_P = \underline{g} - \underline{\omega}_e \times \left(\underline{\omega}_e \times \underline{R}\right) \tag{5.4.1-1}$$

where

$\underline{g}$ = Gravity vector caused by mass attraction as expressed by Equations (5.4-1) and (5.4-2).

$\underline{g}_P$ = Plumb-bob gravity.

$\underline{\omega}_e$ = Earth's rotation rate vector.

$\underline{R}$ = Position location vector from earth center (as in Figure 5.2-1 of Section 5.2).

The $\underline{\omega}_e \times \left(\underline{\omega}_e \times \underline{R}\right)$ term in (5.4.1-1) is evaluated as follows. We first note that $\underline{R}$ can be defined as the sum of two components, a component in the equatorial plane, and a component along the earth polar axis. The cross-product of $\underline{\omega}_e$ in (5.4.1-1) with the earth polar axis component of $\underline{R}$ is zero. Therefore, we can write:

$$\underline{\omega}_e \times \left(\underline{\omega}_e \times \underline{R}\right) = \underline{\omega}_e \times \left(\underline{\omega}_e \times \underline{R}_{Eq}\right) \qquad (5.4.1\text{-}2)$$

where

$\underline{R}_{Eq}$ = Component of $\underline{R}$ in earth's equatorial plane.

From Figure 5.2-1 it should be apparent that from the Section 5.0 definition of earth E Frame coordinates:

$$\underline{R}_{Eq}^E = \left(R_{XE},\, 0,\, R_{ZE}\right)^T$$

or with (5.2.2-1) of Section 5.2:

$$\underline{R}_{Eq}^E = \left(R_S^{\prime} + h\right)\left(u_{Up_{XE}},\, 0,\, u_{Up_{ZE}}\right)^T \qquad (5.4.1\text{-}3)$$

The E Frame components of the earth rate vector $\underline{\omega}_e$ in Equation (5.4.1-2) are from Equation (4.1.1-4):

$$\underline{\omega}_e^E = \left[\ 0\ \ \omega_e\ \ 0\ \right]^T \qquad (5.4.1\text{-}4)$$

where

ω_e = Magnitude of $\underline{\omega}_e$ (See Table 5.6-1 for numerical value).

Substituting (5.4.1-3) and (5.4.1-4) in (5.4.1-2) yields:

$$\underline{\omega}_e \times \left(\underline{\omega}_e \times \underline{R}\right) = -\left(R_S^{\prime} + h\right)\omega_e^2\left(u_{Up_{XE}},\, 0,\, u_{Up_{ZE}}\right)^T \qquad (5.4.1\text{-}5)$$

The North component of (5.4.1-5) is obtained by taking the E Frame dot product with a unit north pointing vector ($\underline{u}_{North}^E$). The $\underline{u}_{North}^E$ vector can be derived as the cross-product between unit vectors up ($\underline{u}_{Up}^E$) and east ($\underline{u}_{East}^E$). An analytical expression for $\underline{u}_{East}^E$ (into the plane of the paper from Figure 5.2-1) can be derived as the normalized cross-product between a unit vector along the earth's polar axis (the Y axis of the E Frame) and $\underline{u}^E_{Up}$. Using Equations (5.1-8) for the $\underline{u}^E_{Up}$ components and $\underline{u}^E_{YE} = (0,\, 1,\, 0)^T$ from the E Frame definition in Section 5.0, the result in the E Frame is:

$$\underline{u}^E_{East} = \frac{\underline{u}^E_{YE} \times \underline{u}^E_{Up}}{\left| \underline{u}^E_{YE} \times \underline{u}^E_{Up} \right|} = \frac{1}{\sqrt{u^2_{Up_{ZE}} + u^2_{Up_{XE}}}} \begin{bmatrix} u_{Up_{ZE}} \\ 0 \\ -\, u_{Up_{XE}} \end{bmatrix} = \frac{1}{\sqrt{1 - u^2_{Up_{YE}}}} \begin{bmatrix} u_{Up_{ZE}} \\ 0 \\ -\, u_{Up_{XE}} \end{bmatrix}$$

$$\underline{u}^E_{North} = \frac{\underline{u}^E_{Up} \times \underline{u}^E_{East}}{\left| \underline{u}^E_{Up} \times \underline{u}^E_{East} \right|} = \frac{1}{\sqrt{1 - u^2_{Up_{YE}}}} \begin{bmatrix} -\, u_{Up_{XE}}\, u_{Up_{YE}} \\ u^2_{Up_{ZE}} + u^2_{Up_{XE}} \\ -\, u_{Up_{ZE}}\, u_{Up_{YE}} \end{bmatrix} \qquad (5.4.1\text{-}6)$$

$$= \frac{1}{\sqrt{1 - u^2_{Up_{YE}}}} \begin{bmatrix} -\, u_{Up_{XE}}\, u_{Up_{YE}} \\ 1 - u^2_{Up_{YE}} \\ -\, u_{Up_{ZE}}\, u_{Up_{YE}} \end{bmatrix}$$

Using (5.4.1-6) for $\underline{u}^E_{North}$, the North component of (5.4.1-5) is then found as:

$$\left[\underline{\omega}^E_e \times \left(\underline{\omega}^E_e \times \underline{R}^E \right) \right] \cdot \underline{u}^E_{North} = -\frac{\left(R'_S + h \right)}{\sqrt{1 - u^2_{Up_{YE}}}}\, \omega^2_e \begin{bmatrix} u_{Up_{XE}} \\ 0 \\ u_{Up_{ZE}} \end{bmatrix} \cdot \begin{bmatrix} -\, u_{Up_{XE}}\, u_{Up_{YE}} \\ 1 - u^2_{Up_{YE}} \\ -\, u_{Up_{ZE}}\, u_{Up_{YE}} \end{bmatrix}$$

$$= \frac{\left(R'_S + h \right)}{\sqrt{1 - u^2_{Up_{YE}}}}\, \omega^2_e\, u_{Up_{YE}} \left(u^2_{Up_{XE}} + u^2_{Up_{ZE}} \right) \qquad (5.4.1\text{-}7)$$

$$= \left(R'_S + h \right) \omega^2_e\, u_{Up_{YE}} \sqrt{1 - u^2_{Up_{YE}}}$$

The vertical component of (5.4.1-5) is found from the dot product with $\underline{u}^E_{Up}$:

$$\left[\underline{\omega}_e^E \times \left(\underline{\omega}_e^E \times \underline{R}^E\right)\right] \cdot \underline{u}_{Up}^E = -\left(R_S' + h\right)\omega_e^2 \begin{bmatrix} u_{Up_{XE}} \\ 0 \\ u_{Up_{ZE}} \end{bmatrix} \cdot \begin{bmatrix} u_{Up_{XE}} \\ u_{Up_{YE}} \\ u_{Up_{ZE}} \end{bmatrix}$$

$$= -\left(R_S' + h\right)\omega_e^2 \left(u_{Up_{XE}}^2 + u_{Up_{ZE}}^2\right) = -\left(R_S' + h\right)\omega_e^2 \left(1 - u_{Up_{YE}}^2\right) \tag{5.4.1-8}$$

After dividing (5.4.1-7) by $\sqrt{1 - u_{Up_{YE}}^2}$ and substitution with (5.4.1-8) into (5.4.1-1), the North and vertical components of plumb-bob gravity $\underline{g}_P$ are obtained below. The East component is zero.

$$\left(\frac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}}\right) = \left(\frac{g_{North}}{\sqrt{1 - u_{Up_{YE}}^2}}\right) - \left(R_S' + h\right)\omega_e^2\, u_{Up_{YE}}$$

$$g_{P_{Up}} = g_{Up} + \left(R_S' + h\right)\omega_e^2 \left(1 - u_{Up_{YE}}^2\right) \tag{5.4.1-9}$$

where

$g_{P_{North}}, g_{P_{Up}}$ = North, Up (vertical) components of plumb-bob gravity $\underline{g}_P$.

$\left(\frac{g_{North}}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$, g_{Up} = Mass attraction gravity terms as calculated in Equations (5.4-4).

in which R_S' is calculated with (5.1-10) and $u_{Up_{YE}}$ is given by (5.2.3-1) or (5.3-16).

The navigation N Frame vertical component (Z) of $\underline{g}_P$ equals $g_{P_{Up}}$ because, from its Section 2.2 definition, the N Frame Z axis is along $\underline{u}_{Up}$. The horizontal X, Y navigation N Frame components of $\underline{g}_P$ are obtained by multiplying $g_{P_{North}}$ by the sine and cosine of the wander angle α relating N Frame X, Y axes to horizontal North/East (X, Y) axes:

$$g_{P_{XN}} = g_{P_{North}} \sin\alpha \qquad g_{P_{YN}} = g_{P_{North}} \cos\alpha \qquad g_{P_{ZN}} = g_{P_{Up}} \tag{5.4.1-10}$$

Substituting (5.3-12) into (5.4.1-10) then yields:

$$g_{P_{XN}} = \left(\frac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}}\right) D_{21} \qquad g_{P_{YN}} = \left(\frac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}}\right) D_{22}$$

$$g_{P_{ZN}} = g_{P_{Up}} \tag{5.4.1-11}$$

The $g_{P_{Up}}$ and $\left(\dfrac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}} \right)$ terms in (5.4.1-11) are calculated with Equations (5.4.1-9) using (5.1-10) for $R_S^{'}$ and (5.3-16) for $u_{Up_{YE}}$. Note how the grouping of terms in (5.4.1-11) has avoided singularities when $u_{Up_{YE}} \pm 1$. As an exercise, the reader is encouraged to trace the $\left(\dfrac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}} \right)$ term back to its origin in Equations (5.4-1) and (5.4-2) to verify that no singularities are encountered in the computation chain. The Table 5.6-1 equation summary (to follow) would be helpful in this regard.

5.5 SURFACE ALTITUDE RATE TERM ANALYSIS

This section provides a rigorous analytical basis for equating the $\underline{u}_{ZN}^{N} \cdot \left(C_E^N \, \dot{\underline{R}}_S^E \right)$ term in Equation (4.4.1.2-7) (Section 4.4.1.2) to zero.

Returning to Equations (5.1-9) of Section 5.1, we first write for $\underline{R}_S^E$:

$$
\begin{aligned}
\underline{R}_S^E &= R_S^{'} \begin{bmatrix} u_{Up_{XE}} \\ (1 - e)^2 \, u_{Up_{YE}} \\ u_{Up_{ZE}} \end{bmatrix} = R_S^{'} \begin{bmatrix} u_{Up_{XE}} \\ u_{Up_{YE}} + \left[(1 - e)^2 - 1 \right] u_{Up_{YE}} \\ u_{Up_{ZE}} \end{bmatrix} \\
&= R_S^{'} \left\{ \underline{u}_{Up}^E + \left[(1 - e)^2 - 1 \right] u_{Up_{YE}} \, \underline{u}_{YE}^E \right\}
\end{aligned} \tag{5.5-1}
$$

where

$\underline{u}_{YE}^E$ = Unit vector along E Frame Y axis which, for the E Frame definition in Section 5.0, lies along the earth's polar axis.

The derivative of (5.5-l) is:

$$
\dot{\underline{R}}_S^E = \dot{R}_S^{'} \left\{ \underline{u}_{Up}^E + \left[(1 - e)^2 - 1 \right] u_{Up_{YE}} \, \underline{u}_{YE}^E \right\} + R_S^{'} \left\{ \dot{\underline{u}}_{Up}^E + \left[(1 - e)^2 - 1 \right] \dot{u}_{Up_{YE}} \, \underline{u}_{YE}^E \right\} \tag{5.5-2}
$$

Multiplying (5.5-2) by C_E^N obtains:

$$
\begin{aligned}
C_E^N \, \dot{\underline{R}}_S^E = {} & \dot{R}_S^{'} \left\{ \underline{u}_{Up}^N + \left[(1 - e)^2 - 1 \right] u_{Up_{YE}} \, \underline{u}_{YE}^N \right\} \\
& + R_S^{'} \left\{ C_E^N \, \dot{\underline{u}}_{Up}^E + \left[(1 - e)^2 - 1 \right] \dot{u}_{Up_{YE}} \, \underline{u}_{YE}^N \right\}
\end{aligned} \tag{5.5-3}
$$

Using Equation (4.4.1.1-5) for $\dot{\underline{u}}_{Up}^{E}$, applying generalized Equation (3.2.1-8), and recognizing from (3.2.1-4) that the transpose of a direction cosine matrix equals its inverse, we find for the $C_E^N\, \dot{\underline{u}}_{Up}^{E}$ term in (5.5-3):

$$C_E^N\, \dot{\underline{u}}_{Up}^{E} \;=\; C_E^N \left(\underline{\rho}^E \times\right) \underline{u}_{Up}^{E} \;=\; C_E^N \left(\underline{\rho}^E \times\right) \left(C_E^N\right)^T C_E^N\, \underline{u}_{Up}^{E} \;=\; \underline{\rho}^N \times \underline{u}_{Up}^{N} \tag{5.5-4}$$

where

$\underline{\rho}^N$ = Transport rate (the angular rate of Frame N relative to Frame E) as projected on Frame N axes.

Equation (5.5-3) thereby becomes:

$$\begin{aligned} C_E^N\, \dot{\underline{R}}_S^{E} \;=\; & \dot{R}_S' \left\{ \underline{u}_{Up}^{N} + \left[(1 - e)^2 - 1 \right] u_{Up_{YE}}\, \underline{u}_{YE}^{N} \right\} \\ & + R_S' \left\{ \underline{\rho}^N \times \underline{u}_{Up}^{N} + \left[(1 - e)^2 - 1 \right] \dot{u}_{Up_{YE}}\, \underline{u}_{YE}^{N} \right\} \end{aligned} \tag{5.5-5}$$

Taking the dot product of (5.5-5) with $\underline{u}_{Up}^{N}$ finds:

$$\underline{u}_{Up}^{N} \cdot \left(C_E^N\, \dot{\underline{R}}_S^{E} \right) \;=\; \dot{R}_S' + \left[(1 - e)^2 - 1 \right] \left(\dot{R}_S'\; u_{Up_{YE}} + R_S'\, \dot{u}_{Up_{YE}} \right) \underline{u}_{Up}^{N} \cdot \underline{u}_{YE}^{N} \tag{5.5-6}$$

The $\dot{R}_S'$ term in (5.5-6) can be developed from the derivative of the square of R_S' in Equation (5.1-10), rearranged:

$$\left\{ 1 + u_{Up_{YE}}^2 \left[(1 - e)^2 - 1 \right] \right\} R_S'^{\,2} \;=\; R_0^2$$

Taking the derivative finds:

$$2\, u_{Up_{YE}}\, \dot{u}_{Up_{YE}} \left[(1 - e)^2 - 1 \right] R_S'^{\,2} + 2 \left\{ 1 + u_{Up_{YE}}^2 \left[(1 - e)^2 - 1 \right] \right\} R_S'\, \dot{R}_S' \;=\; 0$$

and with rearrangement:

$$\dot{R}_S' \;=\; \frac{- u_{Up_{YE}}\, \dot{u}_{Up_{YE}} \left[(1 - e)^2 - 1 \right] R_S'}{1 + u_{Up_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \tag{5.5-7}$$

Using (5.5-7), the $\dot{R}_S'\; u_{Up_{YE}} + R_S'\, \dot{u}_{Up_{YE}}$ term in (5.5-6) becomes:

$$\dot{R}_S' \, u_{Up_{YE}} + R_S' \, \dot{u}_{Up_{YE}} = \frac{R_S' \, \dot{u}_{Up_{YE}}}{1 + u_{Up_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \tag{5.5-8}$$

Furthermore, from the definition of $u_{Up_{YE}}$ and using (3.1.1-29):

$$\underline{u}_{Up}^N \cdot \underline{u}_{YE}^N = \underline{u}_{Up}^E \cdot \underline{u}_{YE}^E = u_{Up_{YE}} \tag{5.5-9}$$

Substituting (5.5-8) and (5.5-9) into (5.5-6) then gives:

$$\underline{u}_{Up}^N \cdot \left(C_E^N \, \dot{\underline{R}}_S^E \right) = \dot{R}_S' + \frac{u_{Up_{YE}} \, \dot{u}_{Up_{YE}} \left[(1 - e)^2 - 1 \right] R_S'}{1 + u_{Up_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \tag{5.5-10}$$

With (5.5-7) for $\dot{R}_S'$ and $\underline{u}_{Up}^N = \underline{u}_{ZN}^N$ by definition, (5.5-10) becomes:

$$\underline{u}_{ZN}^N \cdot \left(C_E^N \, \dot{\underline{R}}_S^E \right) = 0 \tag{5.5-11}$$

Equation (5.5-11) provides the analytical justification for equating the $\underline{u}_{ZN}^N \cdot \left(C_E^N \, \dot{\underline{R}}_S^E \right)$ term to zero in Equation (4.4.1.2-7) of Section 4.4.1.2.

5.6 EARTH RELATED NAVIGATION PARAMETER SUMMARY

Table 5.6-1 is a listing of the principal earth related navigation parameter equations from Chapter 5 that would be utilized in strapdown inertial navigation system software packages. Table 5.6-1 lists the algorithm function, input parameters, output parameters and equation number. Definitions for the input/output parameters can be found by cross-referencing through the Parameter Index in the back of the book. Included in Table 5.6-1 are numerical values for earth's shape, gravity and angular rate constants taken from Reference 4, Tables 3.1 and 5.1. The ellipticity e in Table 5.6-1 is identical to the "flattening" parameter f in Reference 4 as defined in Reference 4, Section 7.4. The μ and R_0 parameters in Table 5.6-1 are identical to the GM and a parameters in Reference 4, Table 3.1, but converted from metric units using a conversion factor of 3.280833333 feet per meter. The J_2 and J_3 coefficients in Table 5.6-1 are identical to the negative of the $C_{2,0}$ and $C_{3,0}$ coefficients in Reference 4 (defined in Reference 4, Table 5.2). The $C_{2,0}$ and $C_{3,0}$ coefficients are calculated from the normalized $\overline{C}_{2,0}$ and $\overline{C}_{3,0}$ values in Reference 4, Table 5.1 using the formulas in Reference 4, Table 5.2 (i.e., $C_{2,0} = \sqrt{5}\,\overline{C}_{2,0}$ and $C_{3,0} = \sqrt{7}\,\overline{C}_{3,0}$). Thus, J_2 and J_3 coefficients for Table 5.6-1 were calculated as $J_2 = -\sqrt{5}\,\overline{C}_{2,0}$ and $J_3 = -\sqrt{7}\,\overline{C}_{3,0}$.

Table 5.6-1 Summary Of Earth Related Navigation Parameter Equations

EQUATION FUNCTION	INPUT	OUTPUT	EQUATION
Earth Shape, Gravity And Angular Rate Constants	Reference 4	$\mu = 1.407635730$ E16 Ft^3/Sec^2, $J_2 = 1.082627$ E-3, $J_3 = -2.5327$ E-6, $R_0 = 2.0925604$ E7 Ft, $\omega_e = 7.2921150$ E-5 Rad/Sec, $e = 1/298.257223563$	
Earth Polar Axis Component Of Geodetic Vertical Unit Vector	C_N^E	$u_{Up_{YE}}$	(5.3-16)
Modified Radial Distance To Earth Surface Location	R_0, e, $u_{Up_{YE}}$	R_S'	(5.1-10)
Radial Distance To Earth Surface Location	R_S', e, $u_{Up_{YE}}$	R_S	(5.2.1-4)
Radial Distance To Navigation Point	R_0, R_S, R_S', h	R	(5.2.1-5)
Cosine Of Range Vector Polar Coordinate Angle	$u_{Up_{YE}}$, e, R, h, R_S'	$\cos\phi$	(5.2.2-3)
Modified Sine Of Range Vector Polar Coordinate Angle	R, h, R_S'	$\left(\dfrac{\sin\phi}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$	(5.2.2-5)
Cosine And Modified Sine Of Difference Between Geocentric And Geodetic Latitudes	R, h, R_S', e, $u_{Up_{YE}}$	$\cos\partial l$, $\left(\dfrac{\sin\partial l}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$	(5.2.3-5)
Local Earth Surface Point Radius Of Curvature In Latitude Direction	R_0, e, R_S'	r_{lS}	(5.2.4-25)
Local Navigation Point Radius Of Curvature In Latitude Direction	r_{lS}, h	r_l	(5.2.4-37)

EQUATION FUNCTION	INPUT	OUTPUT	EQUATION
N Frame Curvature Matrix	r_l, C_N^E, h, R_S', e	F_C^N	(5.3-18)
Vertical Transport Rate Component	Section 4.5 For Options	ρ_{ZN}	Section 4.5 For Options
Unit Vector Upward In N Frame	Definition	$\underline{u}_{ZN}^N$	(5.3-18)
N Frame Transport Rate Vector	$\underline{v}^N$, F_C, ρ_{ZN}, $\underline{u}_{ZN}^N$	$\underline{\rho}^N = \underline{\omega}_{EN}^N$	(5.3-17)
Gravity Components In Polar Coordinates	μ, R, R_0, cos ϕ, J_2, J_3, ...	g_r, $\left(\dfrac{g_\phi}{\sin\phi}\right)$, g_θ	(5.4-1) &(5.4-2)
North And Vertical Gravity Components	g_r, $\left(\dfrac{g_\phi}{\sin\phi}\right)$, $u_{Up\,YE}$, cos ∂l, $\left(\dfrac{\sin\partial l}{\sqrt{1-u_{Up\,YE}^2}}\right)$, $\left(\dfrac{\sin\phi}{\sqrt{1-u_{Up\,YE}^2}}\right)$	$\left(\dfrac{g_{North}}{\sqrt{1-u_{Up\,YE}^2}}\right)$, g_{Up}	(5.4-4)
North And Vertical Plumb-bob Gravity Components	$\left(\dfrac{g_{North}}{\sqrt{1-u_{Up\,YE}^2}}\right)$, g_{Up}, R_S', h, $u_{Up\,YE}$, ω_e	$\left(\dfrac{g_{P_{North}}}{\sqrt{1-u_{Up\,YE}^2}}\right)$, $g_{P_{Up}}$	(5.4.1-9)
N Frame Plumb-bob Gravity Components	$\left(\dfrac{g_{P_{North}}}{\sqrt{1-u_{Up\,YE}^2}}\right)$, $g_{P_{Up}}$, C_N^E	$\underline{g}_P^N$	(5.4.1-11)

6 Quasi-Stationary Initialization

6.0 OVERVIEW

The basic computational process in a strapdown inertial navigation system consists of the integration of attitude, velocity and position rate equations which must first be initialized at the start of navigation. In many applications, the initialization is performed under "quasi-stationary" conditions. Quasi-stationary conditions are characterized as having bounded position and attitude movement such as produced by wind gusts and passenger, fuel, stores loading for an airplane on the ground with parking brake engaged. The knowledge that quasi-stationary conditions apply, allows the attitude and velocity initialization process to be performed autonomously within the inertial navigation system (i.e., without attitude or velocity inputs) through software operations on the system's inertial sensor signals. Position initialization requires external latitude, longitude, altitude (or equivalent) external inputs.

This chapter discusses the principal analytical operations typically implemented for the quasi-stationary initialization process. This includes initialization of the sensor assembly attitude (B Frame orientation relative to the L Frame as represented by the C_B^L matrix or q_B^L attitude quaternion), velocity (represented in the N Frame by $\underline{v}^N$), and the earth referenced position location (altitude h and angular orientation of the N Frame relative to the E Frame as represented by the C_N^E matrix). The primary coordinate frames utilized in this chapter are the E, N, L, Geo and B Frames defined in Section 2.2, where the E Frame axes are further specified as:

E = Earth fixed coordinate frame having its Y axis along the earth's polar axis with X and Z axes in the equatorial plane. The E Frame Z Axis is defined to be parallel with the intersection of the Greenwich reference meridian plane (contains Greenwich England) and the equatorial plane.

Other specialized coordinate frames are defined in the sections in which they are applied.

6.1 ATTITUDE (FRAME B TO FRAME L) INITIALIZATION

The basic method used to initialize the C_B^L direction cosine matrix is based on dynamically observing and correcting C_B^L through the components of $\underline{a}_{SF}^N$ calculated by Equations (4.2-1) and (4.2-3) with (4.1-1) and (4.1.1-1), repeated below for easy reference:

$$\begin{aligned}
\dot{C}_B^L &= C_B^L \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IL}^L \times\right) C_B^L \\
\underline{\omega}_{IL}^L &= C_N^L \left(\underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N\right) \\
\underline{a}_{SF}^L &= C_B^L \, \underline{a}_{SF}^B \\
\underline{a}_{SF}^N &= C_L^N \, \underline{a}_{SF}^L
\end{aligned} \tag{6.1-1}$$

For a quasi-stationary initial alignment, the average value for velocity $\underline{v}^N$ and its derivative $\dot{\underline{v}}^N$ will be zero. From the $\dot{\underline{v}}^N$ expression in (4.3-18), zero average $\dot{\underline{v}}^N$ and $\underline{v}^N$ corresponds to the average specific force acceleration $\underline{a}_{SF}$ being equal to the negative of the local plumb-bob gravity vector $\underline{g}_P$. The initialization of C_B^L is based on aligning the L Frame Z axis with the local plumb-bob gravity vertical which, as discussed in Section 12.1.1 (following Equation (12.1.1-8)), is almost exactly along the geodetic vertical. From the previous discussion we see that this is equivalent to setting C_B^L such that $\underline{a}_{SF}^L$ as calculated with Equations (6.1-1) lies along the negative L Frame Z axis (i.e., has zero X, Y horizontal components). This is typically achieved using a two-step process; Coarse Leveling followed by Fine Alignment.

Coarse Leveling rapidly erects the C_B^L matrix to an approximate vertical alignment such that during Fine Alignment, first order approximations can be safely applied for remaining residual verticality errors. Fine Alignment brings the C_B^L matrix verticality to an accuracy sufficient to initiate inertial navigation operations (the "Navigation Mode"). Fine Alignment also determines the heading of the C_B^L matrix (relative to true North) which is used to initialize the wander angle of the C_N^E matrix or alternatively, to adjust the C_B^L matrix to a desired initial wander angle setting (e.g., zero). The following subsections describe Coarse Leveling and Fine Alignment operations in a quasi-stationary environment.

6.1.1 COARSE LEVELING

Coarse Leveling is a process for rapidly initializing C_B^L to an approximate vertical L Frame attitude. As discussed in Section 6.1, the concept is based on the average $\underline{a}_{SF}$ specific force acceleration being equal to - $\underline{g}_P$ and the L Frame Z axis being along $\underline{g}_P$. With the understanding that gravity acts downward and the definition of the L Frame in Section 2.2, we can then write:

$$\underline{u}_{ZL} = \underline{g}_P / g_P \tag{6.1.1-1}$$

where

$\underline{u}_{ZL}$ = Unit vector along the L Frame Z axis.

g_P = Magnitude of $\underline{g}_P$.

With $\underline{a}_{SF}$ approximately equal to - $\underline{g}_P$ under quasi-stationary conditions, (6.1.1-1) becomes in the B Frame:

$$\underline{u}_{ZL}^{B} \approx - \underline{a}_{SF}^{B} / a_{SF} \tag{6.1.1-2}$$

where

a_{SF} = Magnitude of $\underline{a}_{SF}^{B}$.

Equations (3.2.1-3) and (3.2.1-6) applied to the B and L Frames shows that:

$$C_B^L = \begin{bmatrix} \left(\underline{u}_{XL}^{B}\right)^T \\ \left(\underline{u}_{YL}^{B}\right)^T \\ \left(\underline{u}_{ZL}^{B}\right)^T \end{bmatrix} \tag{6.1.1-3}$$

where

$\underline{u}_{XL}^{B}, \underline{u}_{YL}^{B}, \underline{u}_{ZL}^{B}$ = Unit vectors along the L Frame X, Y, Z axes projected on B Frame axes.

Thus, with Equation (6.1.1-2) for $\underline{u}_{ZL}^{B}$, we can initialize the third row of C_B^L as the negative transpose of the normalized specific force acceleration vector sensed in B Frame body axes by the strapdown accelerometers. Because Equation (6.1.1-2) is an approximation, this initialization operation will result in an approximate “leveling” of C_B^L, hence the terminology “Coarse Leveling”. The actual implementation of Equation (6.1.1-2) typically entails an averaging of $\underline{a}_{SF}^{B}$ over a short time period (e.g., one half second). The form of (6.1.1-2) assures that the resulting third row of C_B^L represents a unit vector.

Once the third row of C_B^L is initialized, the setting of rows one and two is somewhat arbitrary, so long as they properly characterize the direction cosine matrix properties of being unity in magnitude and orthogonal to each other and row three (i.e., the properties represented by Equation (6.1.1-3)).

A simple specification for defining the second row of C_B^L is that the column one component (i.e., C_{21} as defined in (4.1-6)) be zero:

$$C_{21} = 0 \tag{6.1.1-4}$$

By this selection we are setting the locally level Y axis of the L Frame to be perpendicular to the X axis of the B Frame (strapdown sensor assembly axes). The advantage in this approach is that for initial attitudes of the B Frame (strapdown sensor assembly axes) with Y or Z axes vertical, C_{21} under stationary conditions becomes a direct measure of Y or Z angular rate sensor error during Fine Alignment (and Navigation). For a wander azimuth N Frame implementation (See Section 4.5 for definition), the angular rate of the N Frame relative to the earth is zero under stationary translational conditions (as is the L Frame angular rate from its Section 2.2 definition). Therefore, for a stationary attitude (i.e., stationary B Frame orientation relative to the earth), the C_B^L matrix should ideally remain constant. Thus, for either the Y or Z axes vertical, C_{21} (the cosine of the angle between the L Frame Y and B Frame X axes) should remain at the (6.1.1-4) initial condition. A C_{21} value differing from zero equals the integrated angular rate sensor error since Coarse Leveling completion. This is a useful relationship for measuring angular rate sensor error in the system test laboratory.

For rows two and three to be perpendicular, their dot product must be zero. From the definition of the C_B^L rows in (4.1-6) we then have:

$$C_{21}\,C_{31} + C_{22}\,C_{32} + C_{23}\,C_{33} = C_{22}\,C_{32} + C_{23}\,C_{33} = 0 \tag{6.1.1-5}$$

Equation (6.1.1-5) is satisfied by:

$$C_{22} = K\,C_{33} \qquad C_{23} = -\,K\,C_{32} \tag{6.1.1-6}$$

where

K = Constant selected to normalize row two (i.e., the sum of the squares of its elements should be unity).

Equating the sum of the squares of the (6.1.1-4) and (6.1.1-6) expressions to unity, we find for K to normalize row two:

$$K = \frac{1}{\sqrt{C_{32}^2 + C_{33}^2}} \tag{6.1.1-7}$$

Equations (6.1.1-4) and (6.1.1-6) with (6.1.1-7) for K then yields for the initial value of row two:

$$C_{21} = 0 \qquad C_{22} = C_{33} / \sqrt{C_{32}^2 + C_{33}^2} \qquad C_{23} = -C_{32} / \sqrt{C_{32}^2 + C_{33}^2} \tag{6.1.1-8}$$

Initialization of row one is trivial once rows two and three are computed. The rows of C_B^L being mutually perpendicular (from (6.1.1-3)) allows row one to be calculated as the cross-product between rows two and three:

$$\begin{aligned} C_{11} &= C_{22}\,C_{33} - C_{23}\,C_{32} \\ C_{12} &= C_{23}\,C_{31} - C_{21}\,C_{33} \\ C_{13} &= C_{21}\,C_{32} - C_{22}\,C_{31} \end{aligned} \tag{6.1.1-9}$$

The above procedure for leveling the C_B^L matrix works as long as the B Frame X axis is not vertical. For the X axis vertical, the B Frame Y and Z axes are perpendicular to the L Frame Z axis, hence, the magnitudes of C_{32} and C_{33} are both zero. Then Equations (6.1.1-8) become indeterminate because both the numerator and denominator of the C_{22} and C_{23} expressions become zero. For the B Frame X axis near vertical, a different set of logic must be used. For example, a $\left| C_{31} \right|$ greater than 0.85 condition can be applied to signal the need for a revised set of Coarse Leveling initialization logic ($\left| C_{31} \right|$ equal to one corresponds to the B Frame X axis being parallel to the L Frame vertical Z axis). For the alternate logic, C_{23} can be set to zero (rather than C_{21} as in (6.1.1-4)) and we proceed as before:

$$\begin{aligned} &C_{23} = 0 \\ &C_{21}\,C_{31} + C_{22}\,C_{32} + C_{23}\,C_{32} = C_{21}\,C_{31} + C_{22}\,C_{32} = 0 \end{aligned} \tag{6.1.1-10}$$

$$C_{21} = K\,C_{32} \qquad C_{22} = -K\,C_{31} \tag{6.1.1-11}$$

Row two thereby becomes:

$$C_{21} = C_{32} / \sqrt{C_{31}^2 + C_{32}^2} \qquad C_{22} = -C_{31} / \sqrt{C_{31}^2 + C_{32}^2} \qquad C_{23} = 0 \tag{6.1.1-12}$$

Equation (6.1.1-9) can be used as before to evaluate the row one components.

It should be noted that for the B Frame X axis vertical (for which C_{31} is greater than 0.85 so that (6.1.1-12) is used for row two initialization), C_{23} becomes a direct measure of integrated B Frame X axis angular rate sensor error since Coarse Leveling completion (i.e., movement of the computed B Frame Z axis relative to the L Frame Y axis). This was the motivation for specifying $C_{23} = 0$ for C_{31} greater than 0.85.

6.1.2 FINE ALIGNMENT

The quasi-stationary Fine Alignment process is designed to precision level the C_B^L matrix in the presence of quasi-stationary disturbances sensed by the strapdown inertial sensors. Additionally, Fine Alignment estimates horizontal earth rate components along N Frame axes which are used at Fine Alignment completion to initialize the C_B^L heading (azimuth) orientation (or, alternatively, the wander angle in the C_N^E matrix).

Fine Alignment is an iterative estimation/filtering process based on observing the double integral of transformed accelerometer measured specific force $\underline{a}_{SF}^N$ calculated from Equations (6.1-1), thereby deducing and correcting residual errors in C_B^L verticality while simultaneously estimating horizontal earth rate. Using the quasi-stationary assumptions defined in Section 6.0 and a wander azimuth L Frame for Fine Alignment (See Section 4.5 for definition), Equations (6.1-1) for the Fine Alignment process can be simplified and expanded to:

$$
\begin{aligned}
\dot{C}_B^L &= C_B^L \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IL}^L \times\right) C_B^L \\
\underline{\omega}_{IL}^L &= C_N^L \, \underline{\omega}_{IE}^N \\
\dot{\underline{v}}_H^N &= \underline{a}_{SF_H}^N = \left(C_L^N \, C_B^L\right)_H \underline{a}_{SF}^B \\
\Delta\dot{\underline{R}}_H^N &= \underline{v}_H^N
\end{aligned}
\qquad (6.1.2\text{-}1)
$$

where

H = Subscript designation for horizontal components of the associated vector.

$\underline{v}^N$ = Velocity relative to the earth.

$\Delta\underline{R}^N$ = Position divergence defined as position movement from the average position location during the quasi-stationary Fine Alignment process.

The $\dot{\underline{v}}_H^N$ expression in Equations (6.1.2-1) is an approximation to the horizontal component of Equation (4.3-18). It is based on the quasi-stationary assumption of zero average velocity and the approximation that $\underline{g}_P$ is vertical, hence, $\underline{g}_P^N$ horizontal components are approximately zero. The $\underline{\omega}_{IL}^L$ expression in Equations (6.1.2-1) has been simplified from its (6.1-1) form based on the use of a wander azimuth L Frame, hence, zero vertical $\underline{\omega}_{EL}^L$ component, and the quasi-stationary zero average velocity assumption, thus, zero average transport rate (horizontal $\underline{\omega}_{EL}^{L}$ component).

Equations (6.1.2-1) are integrated during Fine Alignment to calculate horizontal position divergence $\Delta\underline{R}_{H}^{N}$ which is used as a measure of the tilt (verticality error) in the C_{B}^{L} matrix. From Equations (6.1.2-1), tilt in C_{B}^{L} is produced by Fine Alignment initialization error (at Coarse Leveling completion), uncertainty in the $\underline{\omega}_{IE}^{N}$ earth rate vector input to the C_{B}^{L} rate equation (through $\underline{\omega}_{IL}^{L}$), and inertial sensor output noise present on the $\underline{\omega}_{IB}^{B}$, $\underline{a}_{SF}^{B}$ inputs. Note that systematic errors in $\underline{\omega}_{IB}^{B}$, $\underline{a}_{SF}^{B}$ will also produce $\Delta\underline{R}_{H}^{N}$ position divergence, however, the Fine Alignment concept is based on ignoring these as negligible (by inertial component design/selection). As a result, residual systematic errors in $\underline{\omega}_{IB}^{B}$, $\underline{a}_{SF}^{B}$ will generate an error in the C_{B}^{L} initialization process which is deemed acceptable.

Based on the previous discussion, Fine Alignment can be analytically defined as the process of using $\Delta\underline{R}_{H}^{N}$ in recursive iterative fashion to estimate and correct tilt (verticality error) in C_{B}^{L} while simultaneously estimating and correcting uncertainties in the earth rate vector $\underline{\omega}_{IE}^{N}$ input to the C_{B}^{L} rate equation. Correcting $\underline{\omega}_{IE}^{N}$ earth rate uncertainties during Fine Alignment process is important for two reasons; 1. Initial uncertainties in $\underline{\omega}_{IE}^{N}$ are too large to enable an accurate leveling of C_{B}^{L}, and 2. The estimated horizontal components of $\underline{\omega}_{IE}^{N}$ are used for determining the L Frame heading attitude relative to the earth (through C_{N}^{E} initialization or adjustment of C_{B}^{L} as discussed in Section 6.2) which completes the C_{B}^{L} initialization process.

Equations (6.1.2-1) are now expanded into the form utilized during Fine Alignment for C_{B}^{L} tilt correction and $\underline{\omega}_{IE}^{N}$ estimation. For this expansion we equate $\underline{\omega}_{IE}^{N}$ to the sum of its vertical and horizontal components, the vertical component equaling earth rate magnitude times the cosine of the angle between the local vertical and earth's rotation axis (i.e., $u_{Up_{YE}}$ in Section 5.2.3 which, by Equation (5.2.3-1), equals the sine of geodetic latitude). The expanded form of (6.1.2-1) then is:

$$
\begin{aligned}
\dot{C}_B^L &= C_B^L \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IL}^L \times\right) C_B^L \\
\underline{\omega}_{IL}^L &= C_N^L \left(\underline{\omega}_{IE}^N + \underline{\omega}_{Tilt}^N\right) \\
\underline{\omega}_{Tilt}^N &= K_2 \, \underline{u}_{ZN}^N \times \Delta\underline{R}_H^N \\
\underline{\omega}_{IE}^N &= \underline{\omega}_{IE_H}^N + \underline{u}_{ZN}^N \, \omega_e \sin l \\
\dot{\underline{\omega}}_{IE_H}^N &= K_1 \, \underline{u}_{ZN}^N \times \Delta\underline{R}_H^N \\
\dot{\underline{v}}_H^N &= \left(C_L^N C_B^L\right)_H \underline{a}_{SF}^B - K_3 \, \Delta\underline{R}_H^N \\
\Delta\dot{\underline{R}}_H^N &= \underline{v}_H^N - K_4 \, \Delta\underline{R}_H^N
\end{aligned}
\qquad (6.1.2\text{-}2)
$$

where

$\underline{u}_{ZN}^N$ = Unit vector along the N Frame vertical axis (Z), projected on N Frame axes.

K_1, K_2, K_3, K_4 = Fine Alignment process estimation feedback control gains.

$\underline{\omega}_{Tilt}^N$ = Angular rate feedback to correct C_B^L tilt.

ω_e = Earth rate magnitude.

l = Geodetic latitude (assumed to be available as an error free input).

The $\underline{u}_{ZN}^N$ cross-product operations in Equations (6.1.2-2) have been introduced as a mathematical method for generating L Frame rotation corrections ($\underline{\omega}_{Tilt}^N$ and $\dot{\underline{\omega}}_{IE_H}^N$) that are ninety degrees rotated about the vertical from $\Delta\underline{R}_H^N$. It can be verified from geometrical reasoning that this is the proper phasing for corrections to L Frame tilt that generated $\Delta\underline{R}_H^N$.

The Fine Alignment process consists of integrating Equations (6.1.2-2) until $\Delta\underline{R}_H^N$ reaches an acceptable quasi-stationary equilibrium. At Fine Alignment completion C_B^L will thereby be accurately leveled and $\underline{\omega}_{IE_H}^N$ accurately estimated. Additionally, the horizontal quasi-stationary velocity and position vectors $\underline{v}_H^N$ and $\Delta\underline{R}_H^N$ will become accurate representations of the true quasi-stationary environmental motion. The initial value for C_B^L at Fine Alignment initiation is its value at Coarse Leveling completion. The Fine Alignment initial value for horizontal earth rate $\underline{\omega}_{IE_H}^N$ is zero (i.e., complete uncertainty). The horizontal velocity $\underline{v}_H^N$ and position divergence $\Delta\underline{R}_H^N$ are also initialized at zero as a best guess (because their actual value is unknown). The K_1, K_2, K_3, K_4 feedback control gains are typically calculated based on Kalman filter theory (See Sections 15.2.1 and 15.2.1.1) which produces time varying gain profiles that are functions of the statistical properties of the Equations (6.1.2-2) computational parameter initial uncertainties, inertial sensor noise, and random disturbance environment characteristics. (Technically, the K_1 - K_4 gains in Equations (6.1.2-2) are continuous form representations of the actual digital discrete Kalman filter gains - See Equation (15.1.2.1-28) for the discrete gain matrix analytical representation. See Equation (15.1.5.3.2-14) of Section 15.1.5.3.2 for the analytical representation of the continuous form Kalman gain matrix).

We note in passing that the purpose for adding the $\underline{v}_H^N$ and $\Delta\underline{R}_H^N$ integrations in Equations (6.1.2-2) as part of the Fine Alignment process is to provide filtering of disturbance and sensor noise in the "forward loop" prior to application of the feedback control gains. From a Kalman filter design standpoint, use of $\Delta\underline{R}_H^N$ allows the "measurement" noise to be modeled as an easily defined quasi-stationary position disturbance (See Section 15.2.1.2).

6.1.3 REMOVAL OF RESIDUAL TILT EFFECTS AT FINE ALIGNMENT COMPLETION

The quasi-stationary Fine Alignment process described in Section 6.1.2 is based on aligning the N (and L) Frame vertical (Z) axis along the plumb-bob gravity vector direction. As discussed in Section 6.1.2, this is a very good approximation to aligning the N Frame Z along the local geodetic vertical (i.e., perpendicular to the local earth reference ellipsoid surface), the proper orientation according to the N frame definition in Section 2.2. The approximation was realized by neglecting the horizontal components of gravity in the Equations (6.1.2-1) - (6.1.2-2) $\dot{\underline{v}}_H^N$ expression (compared with the correct Equation (4.3-18) horizontal component for $\dot{\underline{v}}_H^N$ that includes $\underline{g}_P^N$ gravity). It is important to realize that including the horizontal component of $\underline{g}_P^N$ in Equations (6.1.2-1) - (6.1.2-2) is not possible because the components of gravity are not known in the N Frame at Fine Alignment initiation. This is due to the fact that gravity is modeled along local North, East, Up axes (see Section 5.4) and cannot be transformed to the N Frame until the N Frame heading relative to the earth has been determined at Fine Alignment completion (from the estimated horizontal earth rate components). At completion of Fine Alignment, however, the N Frame heading is known, hence, $\underline{g}_P^N$ can be calculated, and an adjustment can be computed and applied to C_B^L verticality and to the $\underline{\omega}_{IE_H}^N$ earth rate estimate to correct for deletion of $\underline{g}_P^N$ during the Fine Alignment process.

To correct C_B^L we need to rotate the L Frame so that the new Z axis orientation lines up with the geodetic vertical. Analytically this can be expressed as:

$$C_B^{L_2} = C_{N_2}^{L_2} \, C_{N_1}^{N_2} \, C_{L_1}^{N_1} \, C_B^{L_1} \tag{6.1.3-1}$$

where

L_1, N_1 = Frame L and N orientations at completion of Fine Alignment.

L_2, N_2 = Frame L and N orientations after application of the verticality correction. Frames L_2 and N_2 are the and L and N Frames to be used for inertial navigation following the initial alignment mode.

$C_{N_1}^{N_2}$ = Direction cosine matrix that transforms vectors from Frame N_1 to Frame N_2.

$C_B^{L_1}, C_B^{L_2}$ = C_B^L at completion of Fine Alignment (1) and after application of the verticality correction (2).

C_N^L, C_L^N = Direction cosine matrices that transform vectors from the N to the L and from the L to the N Frames (C_N^L is defined in Equation (4.1.1-2)).

$C_{N_2}^{L_2}, C_{L_1}^{N_1}$ = C_N^L, C_L^N at completion of Fine Alignment (1) and after application of the verticality correction (2).

From the Section 2.2 definition for the L and N Frames, they are fixed relative to one another, hence, $C_{N_2}^{L_2} = C_N^L$ and $C_{L_1}^{N_1} = C_L^N$, and (6.1.3-1) becomes:

$$C_B^{L_2} = C_N^L \, C_{N_1}^{N_2} \, C_L^N \, C_B^{L_1} \tag{6.1.3-2}$$

The $C_{N_1}^{N_2}$ matrix in (6.1.3-2) relates the N Frames before and after the verticality correction. Before the correction, by virtue of the Fine Alignment process, the vertical Z axis of the N_1 Frame becomes aligned to the negative of $\underline{g}_P$. After the verticality correction, the $\underline{g}_P$ components in Frame N_2 should equal the true $\underline{g}_P^N$ components. The $\underline{g}_P$ components in Frames N_1 and N_2 are related by $C_{N_2}^{N_1}$, the transpose of $C_{N_1}^{N_2}$ in (6.1.3-2). Analytically:

$$\underline{g}_P^{N_1} = - g_P \, \underline{u}_{ZN}^N \qquad \underline{g}_P^{N_2} = \underline{g}_P^N \qquad \underline{g}_P^{N_1} = C_{N_2}^{N_1} \, \underline{g}_P^{N_2} \tag{6.1.3-3}$$

where

g_P = Magnitude of $\underline{g}_P$.

or in combination:

$$- g_P \, \underline{u}_{ZN}^N = C_{N_2}^{N_1} \, \underline{g}_P^N \tag{6.1.3-4}$$

Equation (6.1.3-4) can be solved for $C_{N_2}^{N_1}$ if we base the solution on the rotation vector associated with $C_{N_2}^{N_1}$ having minimum magnitude. Generalized Equations (3.2.1.1-1), (3.2.1.1-22) and (3.2.1.1-26) with (3.1.1-12) and (3.1.1-13) show that the solution is given by:

$$C_{N_2}^{N_1} = I + \left(\underline{E}\times\right) + \frac{1}{1 + D} \left(\underline{E}\times\right)^2 \tag{6.1.3-5}$$

with

$$\underline{E} = - \frac{1}{g_P} \, \underline{g}_P^N \times \underline{u}_{ZN}^N \qquad D = - \frac{1}{g_P} \, \underline{g}_P^N \cdot \underline{u}_{ZN}^N \tag{6.1.3-6}$$

The $C_{N_1}^{N_2}$ matrix in (6.1.3-2) is the transpose of (6.1.3-5), or since the transpose of a cross-product operator equals its negative:

$$C_{N_1}^{N_2} = I - \left(\underline{E}\times\right) + \frac{1}{1 + D} \left(\underline{E}\times\right)^2 \tag{6.1.3-7}$$

The $\underline{g}_P^N$ vector in (6.1.3-6) is computed with Equations (5.4.1-9) and (5.2.3-1) and the fact that the Fine Alignment calculated horizontal earth rate component $\underline{\omega}_{IE_H}^N$ lies (approximately) North:

$$\underline{g}_P^N = C_{Geo}^N \, \underline{g}_P^{Geo} \approx C_{Geo}^{N_1} \, \underline{g}_P^{Geo}$$

$$C_{Geo}^{N_1} = \begin{bmatrix} \cos \alpha_1 & \sin \alpha_1 & 0 \\ - \sin \alpha_1 & \cos \alpha_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \underline{g}_P^{Geo} = \begin{bmatrix} 0 \\ \left(\dfrac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}} \right) \cos l \\ g_{P_{Up}} \end{bmatrix} \tag{6.1.3-8}$$

(Continued)

$$\underline{u}^{N_1}_{North} = \underline{\omega}^{N_1}_{IE_{H/1}} / \omega_{IE_{H/1}}$$

$$\cos\alpha_1 = \underline{u}^{N_1}_{North} \cdot \underline{u}^{N_1}_{YN_1} \qquad \sin\alpha_1 = \underline{u}^{N_1}_{North} \cdot \underline{u}^{N_1}_{XN_1} \tag{6.1.3-8 (Continued)}$$

where

Geo = Local geographic North, East, Up coordinates as defined in Section 2.2.

C^{N}_{Geo}, $C^{N_1}_{Geo}$ = Direction cosine matrices that transform vectors from the Geo Frame to the N Frame and from the Geo Frame to the N_1 Frame.

l = Geodetic latitude.

$u_{Up_{YE}}$ = Projection of a geodetic vertical unit vector on the earth's polar axis.

α_1 = Wander angle between the N_1 Frame Y axis and North.

$g_{P_{North}}$, $g_{P_{Up}}$ = North, Up components of $\underline{g}_P$. $\left(\dfrac{g_{P_{North}}}{\sqrt{1 - u^2_{Up_{YE}}}}\right)$ and $g_{P_{Up}}$ are calculated with Equations (5.4.1-9).

$\underline{\omega}^{N_1}_{IE_{H/1}}$ = Horizontal component of the $\underline{\omega}_{IE}$ earth rate vector along the N_1 Frame horizontal projected on N_1 Frame axes (i.e., the $\underline{\omega}_{IE}$ determined at Fine Alignment completion prior to application of the verticality correction).

$\omega_{IE_{H/1}}$ = Magnitude of $\underline{\omega}^{N_1}_{IE_{H/1}}$.

$\underline{u}^{N_1}_{North}$ = Unit vector in the North direction as determined in the N_1 Frame using $\underline{\omega}^{N_1}_{IE_{H/1}}$.

$\underline{u}^{N_1}_{XN_1}$, $\underline{u}^{N_1}_{YN_1}$ = Unit vectors along N_1 Frame X, Y axes, projected on the N_1 Frame.

At completion of Fine Alignment, the C^L_B matrix is corrected for verticality error by applying Equation (6.1.3-2) with (6.1.3-6) - (6.1.3-8). Note that Equation (6.1.3-7) can be simplified by taking advantage of the smallness of $\underline{E}$ that permits the $(\underline{E}\times)^2$ term to be neglected. This approximation is not recommended by the author on grounds of preserving the orthogonality/normality characteristics of C^L_B to precision in applications when the enhancements presented in this section are warranted.

The estimated horizontal earth rate $\underline{\omega}_{IE_{H/1}}^{N_1}$ can also be corrected at Fine Alignment completion by transformation of $\underline{\omega}_{IE_{H/1}}^{N_1}$ into the revised verticality corrected N Frame. Using the (6.1.2-2) $\underline{\omega}_{IE}^{N}$ format finds:

$$\underline{\omega}_{IE_{H/2}}^{N_2} = \left[C_{N_1}^{N_2} \left(\underline{\omega}_{IE_{H/1}}^{N_1} + \underline{u}_{ZN_1}^{N_1} \, \omega_{IE_{ZN/1}} \right) \right]_H \tag{6.1.3-9}$$

where

H = Designator for horizontal components which equal the X, Y components of the parameter in [ ] brackets.

$\underline{\omega}_{IE_{H/2}}^{N_2}$ = Horizontal component of the $\underline{\omega}_{IE}$ earth rate vector along the N_2 Frame horizontal projected on N_2 Frame axes (i.e., the $\underline{\omega}_{IE}$ determined at Fine Alignment completion after application of the verticality correction).

$\omega_{IE_{ZN/1}}$ = Component of $\underline{\omega}_{IE}$ along the Z axis of Frame N_1.

$\underline{u}_{ZN_1}^{N_1}$ = Unit vector along the Z axis of the N_1 Frame as viewed in the N_1 Frame.

Based on the smallness of $\underline{E}$, we can approximate $C_{N_1}^{N_2}$ in (6.1.3-7) as:

$$C_{N_1}^{N_2} \approx I - \left(\underline{E} \times \right) \tag{6.1.3-10}$$

Substituting (6.1.3-10) into (6.1.3-9) then obtains after expansion:

$$\begin{aligned} \underline{\omega}_{IE_{H/2}}^{N_2} &= \left\{ \left[I - \left(\underline{E} \times \right) \right] \left(\underline{\omega}_{IE_{H/1}}^{N_1} + \underline{u}_{ZN_1}^{N_1} \, \omega_{IE_{ZN/1}} \right) \right\}_H \\ &= \underline{\omega}_{IE_{H/1}}^{N_1} + \left(\underline{u}_{ZN_1}^{N_1} \, \omega_{IE_{ZN/1}} \right)_H - \left[\left(\underline{E} \times \right) \underline{\omega}_{IE_{H/1}}^{N_1} \right]_H - \left[\left(\underline{E} \times \right) \underline{u}_{ZN_1}^{N_1} \, \omega_{IE_{ZN/1}} \right]_H \end{aligned} \tag{6.1.3-11}$$

From (6.1.3-6) we see that $\underline{E}$ is perpendicular to $\underline{u}_{ZN}^{N}$, hence, has no Z component. Therefore, since $\underline{\omega}_{IE_{H/1}}^{N_1}$ has no Z component, $\left(\underline{E} \times \right) \underline{\omega}_{IE_{H/1}}^{N_1}$ has no X, Y components, and the $\left[\left(\underline{E} \times \right) \underline{\omega}_{IE_{H/1}}^{N_1} \right]_H$ term in (6.1.3-11) is zero. Because $\underline{u}_{ZN_1}^{N_1}$ has only a Z component (i.e., no X, Y components), $\left(\underline{u}_{ZN_1}^{N_1} \, \omega_{IE_{ZN/1}} \right)_H$ in (6.1.3-11) is also zero. Additionally, because $\underline{u}_{ZN_1}^{N_1}$ has only a Z component, $\left(\underline{E} \times \right) \underline{u}_{ZN_1}^{N_1}$ has only X, Y components. Thus, the horizontal H designation on

$\left[(\underline{E}\times)\, \underline{u}_{ZN_1}^{N_1}\, \omega_{IE_{ZN/1}}\right]_H$ in (6.1.3-11) is redundant and can be removed. Making these substitutions in (6.1.3-11) then yields the correction equation for $\underline{\omega}_{IE_{H/1}}^{N_1}$:

$$\underline{\omega}_{IE_{H/2}}^{N_2} = \underline{\omega}_{IE_{H/1}}^{N_1} - (\underline{E}\times)\, \underline{u}_{ZN_1}^{N_1}\, \omega_{IE_{ZN/1}} \tag{6.1.3-12}$$

with $\underline{E}$ provided by (6.1.3-6).

6.2 NAVIGATION FRAME INITIALIZATION (FRAME N TO FRAME E)

As discussed in Section 6.1.2, initialization of the L Frame azimuth (i.e., angular orientation around the local vertical) relative to the earth E Frame at Fine Alignment completion is based on application of $\underline{\omega}_{IE_H}^{N}$ generated during Fine Alignment. For a strapdown inertial navigation system, L Frame azimuth initialization can be accomplished in either of two ways; wander angle setting in the C_N^E matrix, or direct azimuth adjustment of the C_B^L matrix. Both methods are discussed below.

6.2.1 INITIALIZATION OF N FRAME BY WANDER ANGLE SETTING IN THE FRAME N TO E MATRIX

Initialization of the C_N^E matrix at Fine Alignment completion is equivalent to initialization of latitude, longitude and wander angle (as should be apparent from the C_N^E components in Equations (4.4.2.1-2)). Latitude and longitude are provided as inputs to the inertial navigation system. Initial setting of the wander angle is based on assuring that the resulting N Frame attitude is consistent with the initial C_B^L matrix setting. In this section we will assume that the initial value for C_B^L is its value at completion of the Equations (6.1.2-2) Fine Alignment dynamic process, and following the Section 6.1.3 tilt adjustment. For a quasi-stationary initialization, the wander angle setting that is consistent with this C_B^L value is determined from the horizontal earth rate estimate $\underline{\omega}_{IE_H}^{N}$ generated during Fine Alignment (including the (6.1.3-12) adjustment if applied).

Application of $\underline{\omega}_{IE_H}^{N}$ to C_N^E initialization is based on recognition that the earth rotation vector is fixed in the E Frame, and for the particular E Frame definition in Section 6.0, lies along the E Frame Y axis. We can, therefore, write:

$$\underline{\omega}_{IE}^{N} = \underline{u}_{YE}^{N}\,\omega_e \qquad (6.2.1\text{-}1)$$

or

$$\underline{u}_{YE}^{N} = \frac{1}{\omega_e}\,\underline{\omega}_{IE}^{N} \qquad (6.2.1\text{-}2)$$

where

$\underline{u}_{YE}^{N}$ = Unit vector along the E Frame Y axis as projected on N Frame axes.

From generalized Equations (3.2.1-3) and (3.2.1-6) we know that:

$$\left(C_N^E\right)^T = \begin{bmatrix} \underline{u}_{XE}^{N} & \underline{u}_{YE}^{N} & \underline{u}_{ZE}^{N} \end{bmatrix} \qquad (6.2.1\text{-}3)$$

where

$\underline{u}_{XE}^{N}, \underline{u}_{YE}^{N}, \underline{u}_{ZE}^{N}$ = Unit vectors along E Frame X, Y, Z axes as projected on N Frame axes.

From Equations (6.2.1-2) and (6.2.1-3) we see that the second row of the C_N^E matrix (the second column of $\left(C_N^E\right)^T$) equals the normalized earth rate vector. If we now substitute the $\underline{\omega}_{IE}^{N}$ expression from Equations (6.1.2-2) into (6.2.1-2) we find that the second row of C_N^E is:

$$\begin{bmatrix} D_{21} \\ D_{22} \\ D_{23} \end{bmatrix} = \underline{u}_{YE}^{N} = \frac{1}{\omega_e}\,\underline{\omega}_{IE_H}^{N} + \underline{u}_{ZN}^{N}\sin l \qquad (6.2.1\text{-}4)$$

Equation (6.2.1-4) is the basis for initialization of the second row of C_N^E following a quasi-stationary Fine Alignment in which $\underline{\omega}_{IE_H}^{N}$ has been estimated. Because of residual errors in $\underline{\omega}_{IE_H}^{N}$ at Fine Alignment completion, direct use of $\underline{\omega}_{IE_H}^{N}$ in Equation (6.2.1-4) would generate a second C_N^E row that was not unity in magnitude (i.e., had normalization error). In order to assure a normalized second row, an alternative form of (6.2.1-4) is used based on the dot product of (6.2.1-4) with itself being one, while recognizing that $\underline{\omega}_{IE_H}^{N}$ is perpendicular to $\underline{u}_{ZN}^{N}$ and the magnitude of $\underline{u}_{ZN}^{N}$ is one:

$$1 = \left(\frac{1}{\omega_e}\,\underline{\omega}_{IE_H}^{N} + \underline{u}_{ZN}^{N}\sin l\right)\cdot\left(\frac{1}{\omega_e}\,\underline{\omega}_{IE_H}^{N} + \underline{u}_{ZN}^{N}\sin l\right) = \frac{\omega_{IE_H}^{2}}{\omega_e^{2}} + \sin^2 l = \frac{\omega_{IE_H}^{2}}{\omega_e^{2}} + 1 - \cos^2 l$$

hence, $$\frac{\omega_{IE_H}^2}{\omega_e^2} = \cos^2 l \qquad \text{or} \qquad \omega_e = \frac{\omega_{IE_H}}{\cos\, l} \tag{6.2.1-5}$$

where

$$\omega_{IE_H} = \text{Magnitude of } \underline{\omega}_{IE_H}^N.$$

Substituting for ω_e in (6.2.1-4) then yields:

$$\begin{bmatrix} D_{21} \\ D_{22} \\ D_{23} \end{bmatrix} = \underline{u}_{YE}^N = \frac{\cos\, l}{\omega_{IE_H}}\,\underline{\omega}_{IE_H}^N + \underline{u}_{ZN}^N \sin\, l \tag{6.2.1-6}$$

Once the second row of C_N^E has been determined from (6.2.1-6) with input latitude (l) and $\underline{\omega}_{IE_H}^N$, Equations (4.4.2.1-2) are applied to extract the sine and cosine of the wander angle:

$$\sin\alpha = D_{21} / \cos\, l \qquad \cos\alpha = D_{22} / \cos\, l \tag{6.2.1-7}$$

Row one of C_N^E is then initialized with (6.2.1-7), input latitude (l) and longitude (L):

$$\begin{aligned} D_{11} &= \cos L \cos\alpha - \sin L \sin l \sin\alpha \\ D_{12} &= -\cos L \sin\alpha - \sin L \sin l \cos\alpha \\ D_{13} &= \sin L \cos l \end{aligned} \tag{6.2.1-8}$$

Equations (4.4.2.1-2) can also be used to initialize the third row of C_N^E, or alternatively, the third row can be obtained (as in Equation (4.4.1.1-4)) as the cross-product between rows one and two:

$$\begin{aligned} D_{31} &= D_{12}\, D_{23} - D_{13}\, D_{22} \\ D_{32} &= D_{13}\, D_{21} - D_{11}\, D_{23} \\ D_{33} &= D_{11}\, D_{22} - D_{12}\, D_{21} \end{aligned} \tag{6.2.1-9}$$

6.2.2 INITIALIZATION OF N FRAME BY DIRECT B TO L FRAME MATRIX MODIFICATION

In this section we will assume that the initial value of C_B^L is modified in azimuth from its value at completion of the Equations (6.1.2-2) dynamic alignment process, following the Section 6.1.3 tilt adjustment. For a quasi-stationary initialization, the C_B^L azimuth angle modification is selected to realign the L Frame to be parallel with local geographic axes so that the L Frame X axis is North, Y axis is East and Z axis remains Down. The N Frame attitude relative to the L Frame is fixed (by the Section 2.2 definition) so that after the C_B^L adjustment, the N Frame X axis will be horizontal East, the Y axis horizontal North with the Z axis Up. As such, based on definition of the wander angle (between the N Frame Y axis and North), the wander angle will be zero after the C_B^L azimuth modification. The horizontal earth rate estimate $\underline{\omega}_{IE_H}^N$ generated during Fine Alignment is used to calculate the C_B^L azimuth modification angle.

We now analytically define the C_B^L modification process as:

$$C_B^{L+} = C_{N+}^{L+} \, C_N^{N+} \, C_L^N \, C_B^L \tag{6.2.2-1}$$

where

C_B^L = Value at Fine Alignment completion, following the Section 6.1.3 tilt adjustment.

L+, N+ = L and N Frames following the C_B^L azimuth modification.

C_B^{L+} = C_B^L value following the azimuth angle modification.

The C_B^{L+} matrix then becomes the initial value for C_B^L at the start of the Navigation Mode.

The N and L Frames maintain their relative attitude after the C_B^L azimuth modification, hence, the C_{N+}^{L+} matrix in (6.2.2-1) is simply:

$$C_{N+}^{L+} = C_L^N \tag{6.2.2-2}$$

The C_N^{N+} matrix in (6.2.2-1) is obtained using the Equations (3.2.1-3) and (3.2.1-6) generalized formulas:

$$\left(C_N^{N+}\right)^T = \left[\; \underline{u}_{XN+}^N \quad \underline{u}_{YN+}^N \quad \underline{u}_{ZN+}^N \; \right] \tag{6.2.2-3}$$

where

$\underline{u}_{XN+}^N, \underline{u}_{YN+}^N, \underline{u}_{ZN+}^N$ = Unit vectors along the N+ Frame X, Y, Z axes as projected on N Frame axes.

Following C_B^L modification, the N+ Frame Y axis will be horizontal North, hence, will lie along the horizontal component of the earth rate vector. Therefore, $\underline{u}_{YN+}^N$ equals the normalized value of $\underline{\omega}_{IE_H}^N$ or:

$$\underline{u}_{YN+}^N = \frac{1}{\omega_{IE_H}} \underline{\omega}_{IE_H}^N \tag{6.2.2-4}$$

Because the C_B^L modification is an azimuth angle change (i.e., about the local vertical), $\underline{u}_{ZN+}^N$ in (6.2.2-3) is:

$$\underline{u}_{ZN+}^N = \underline{u}_{ZN}^N \tag{6.2.2-5}$$

The $\underline{u}_{XN+}^N$ unit vector is then calculated from the cross-product between $\underline{u}_{YN+}^N$ and $\underline{u}_{ZN+}^N$ to complete the orthogonal N+ Frame axis unit vector triad:

$$\underline{u}_{XN+}^N = \underline{u}_{YN+}^N \times \underline{u}_{ZN+}^N \tag{6.2.2-6}$$

Application of Equation (6.2.2-1) with (6.2.2-2) - (6.2.2-6) reorients the L and N Frames so that the New N Frame Y axis (and L Frame X axis) points North. Consequently, the associated wander angle is zero, and initialization of the C_N^E matrix using Equations (4.4.2.1-2) becomes:

$$C_N^E = \begin{bmatrix} \cos L & -\sin L \sin l & \sin L \cos l \\ 0 & \cos l & \sin l \\ -\sin L & -\cos L \sin l & \cos L \cos l \end{bmatrix} \tag{6.2.2-7}$$

6.3 VELOCITY INITIALIZATION

Under quasi-stationary conditions, the velocity vector $\underline{v}^N$ at Navigation Mode entry can be initialized to zero. Alternatively, and more accurately, the horizontal $\underline{v}^N$ components can be initialized by setting them to $\underline{v}_H^N$ at Fine Alignment completion as calculated in the Equations (6.1.2-2) dynamic process. A more accurate initialization of the $\underline{v}^N$ vertical component can be realized by implementation of the vertical control loop channel (defined in Equations (4.4.1.2.1-1) - (4.4.1.2.1-3)) during Fine Alignment. At completion of Fine Alignment, the vertical velocity so calculated (as well as the e_{VC_3} integral controller value) would be used as initial Navigation Mode values for these parameters. If this approach is taken, the Equation (4.4.1.2.1-1) vertical channel angular rate cross product terms with horizontal velocity would typically be excluded during Fine Alignment as negligible, and to avoid coupling horizontal velocity error transients into vertical velocity. Initialization of the Equations (4.4.1.2.1-1) - (4.4.1.2.1-3) vertical channel at the start of Fine Alignment would then typically entail setting vertical velocity to zero, the e_{vc_3} integral controller to zero, and altitude h to the pressure altitude input h_{Prsr}. In order to accelerate vertical channel convergence during Fine Alignment, a faster set of control gains (C_1, C_2, C_3) can be applied with the h_{Prsr} input signal clamped at its starting value to eliminate h_{Prsr} noise inputs.

6.4 ALTITUDE INITIALIZATION

Altitude is typically initialized for the Navigation Mode at input altitude. If a vertical channel control loop is used during inertial navigation (as discussed in Section 6.3), altitude can be initialized for the Navigation Mode at the value calculated by Equations (4.4.1.2.1-1) - (4.4.1.2.1-3) at Fine Alignment completion.

7 Strapdown Inertial Navigation Digital Integration Algorithms

7.0 OVERVIEW

The basic functions executed in a strapdown INS computer are the integration of INS angular rate sensor data into attitude (denoted as "attitude integration"), use of the attitude data to transform INS accelerometer data from sensor coordinates (the B Frame) into navigation coordinates (the N Frame), integration of the N Frame acceleration into velocity (denoted as "velocity integration"), and integration of the N Frame velocity into position (denoted as "position integration"). Chapters 4 and 6 described these basic software functions in the form of continuous differential equations. Time-wise integration of the differential equations in the classical sense would provide continuous measurements of INS attitude, velocity and position location. In an actual INS, the integration functions are executed with digital algorithms operating at a specified repetition rate. This chapter uses a comprehensive design process to develop strapdown digital integration algorithms based on the Chapter 4 differential equations. The derived algorithms are designed to generate the identical solution at their update times as would a continuous integration of the Chapter 4 equations. The material presented emphasizes a rigorous analytical formulation and the use of exact closed-form equations, when possible, for ease in computer software documentation/validation (which is also consistent with modern day flight computer technology). Included in the algorithm design process is a rigorous treatment of methods for accounting for navigation coordinate frame rotation during the integration update time periods. An abbreviated version of this chapter was published in 1998 as References 34 and 35.

From a historical perspective, since the basic strapdown inertial navigation concept was originally formulated in the 1950's, strapdown analysts have primarily focused on the design of algorithms for the attitude integration function. Invariably, the design approaches were driven by the capabilities and limitations of contemporary flight computer technology. In the late 1950's and in the 1960's, two approaches were pursued by strapdown analysts (in various organizations) for the angular rate into attitude integration function (References 9, 20, 23, 39, and 40); high speed attitude updating (e.g., 10 - 20 KHz) using first order digital algorithms, and lower speed (e.g., 50 - 100 Hz) attitude updating using higher order algorithms. The high speed approach was promoted as a means for accurately accounting for high frequency angular rate components that can rectify (i.e., in the electronics sense of an oscillating signal rectifying into a constant signal), producing a systematic three-dimensional attitude change. However, computer technology of that time period was only capable of executing simplified first order equations of limited accuracy for the attitude updating algorithms. In contrast, the higher order algorithm proponents touted improved analytical accuracy compared to first order algorithms; however, the improved accuracy was degraded due to the associated increase in executable operations per attitude update cycle, hence a slower attitude update rate to satisfy contemporary computer throughput limitations. Tradeoffs between the two approaches were clouded by the emergence of the attitude quaternion as the "preferred approach" for the analytical form of the computed attitude parameter (versus the traditional direction cosine matrix attitude representation). For the algorithms investigated during that time period, the quaternion showed improved accuracy in high frequency angular rate environments.

In 1966, the writer proposed a new "two-speed" approach for the attitude integration function (Reference 29) by which the attitude updating operation was divided into two parts; a simple high speed first order algorithm portion coupled with a more complex moderate speed higher order algorithm portion whose input was provided by the high speed algorithm. The simplified high speed algorithm accounted for small amplitude high frequency angular oscillations within the attitude update cycle that can rectify into systematic attitude build-up (traditionally denoted as "coning"). The moderate speed higher order algorithm accurately accounted for larger amplitude angular motion over the moderate speed update cycle time period. Taken together, the combined accuracy of the two-speed approach was equivalent to operating the higher order algorithm at the high speed rate (for improved accuracy), however, due to the simplicity of the high speed algorithm, the combined computer throughput requirement was no greater than for original high speed first order attitude updating algorithms. The utility of the Reference 29 two speed algorithm design approach was limited by its basic analytical formulation as a Picard type recursive integration (Reference 19a) of the continuous form attitude rate differential equation in which both the moderate and high speed algorithms were generated simultaneously. The complexity of the analytical recursive integration design process limited expansion of the higher order moderate speed algorithm (to only second order in Reference 29 which was considered acceptable at that time period).

In an unrelated design activity, Jordan (Reference 14) in 1969 suggested a two-speed approach for the strapdown attitude updating function in which the analytical formulation at the onset was based on two separately defined calculations; a moderate speed classical closed-form ("exact") higher order attitude updating algorithm based on input attitude change, and a simplified high speed second order integration algorithm that measured the attitude change input for the moderate speed algorithm. In 1971, Bortz (Reference 2) extended the Jordan concept to have the high speed calculation based on a differential equation that, when integrated, measures the exact attitude change input to the exact attitude updating algorithm. The exact moderate speed attitude algorithm can be structured to any specified order of accuracy by truncation of two trigonometric coefficients. In practice, simplified forms of the Bortz attitude change differential equation have been used for the high speed function. References 2 and 14 thereby provided a more general form of the two-speed attitude updating approach in which the moderate speed higher order algorithm and high speed simplified algorithm can be independently tailored to meet particular application requirements (Interestingly, Reference 2 proposed an analog implementation for a simplified version of the high speed algorithm). A secondary benefit derived from the Reference 2 and 14 two-speed approach (proposed using direction cosines for the exact moderate speed attitude update operation) is that the moderate speed portion can also be formulated with an analytically exact closed-form quaternion updating algorithm using the identical high speed input applied for direction cosine updating. Thus, the new two-speed approach has equal accuracy for either direction cosine or quaternion updating, both of which derive from analytically exact closed-form equations (assuming that Taylor series expansion for trigonometric coefficients is carried out to comparable accuracy order).

Many modern day strapdown inertial navigation systems utilize attitude updating algorithms based on a two speed approach. The repetition rate for the moderate speed algorithm portion (e.g., 50 - 200 Hz) is typically designed, based on maximum angular rate considerations, to minimize power series truncation error effects in the moderate and high speed algorithms. The repetition rate for the high speed algorithm (e.g., 1 - 4 KHz for an aircraft INS with 1 nautical-mile-per-hour 50 percentile radial position error rate) is designed, based on the anticipated strapdown inertial sensor assembly vibration environment, to accurately account for vibration induced coning effects. Continuing two-speed attitude algorithm development work has centered on variations for the high speed integration function. Originally conceived as a simple first order algorithm (Reference 29), today's high speed attitude algorithms have taken advantage of increased throughput capabilities in modern day computers and become higher order for improved accuracy (References 11, 12, 33, and 34). While the attitude updating function has been evolving to its current form, very little parallel work has been published on the development of the companion strapdown INS algorithms for acceleration transformation/integration into velocity and velocity integration into position.

The acceleration transformation algorithm must account for attitude rotation during the acceleration transformation/velocity update period. In some applications, this is achieved using a centering algorithm (Reference 19) in which attitude data for the acceleration transformation is updated at the center of the time interval used for integrated B Frame acceleration increment accumulation (thereby introducing a staggered attitude-update/velocity-update integration architecture). A variation of this approach updates the attitude at twice the velocity update rate, with the attitude solution between velocity updates used for transformation of the integrated B Frame acceleration increment measured over the velocity update cycle. A two-speed approach can also be used for acceleration-transformation/velocity-integration in a dynamic environment that parallels the two-speed attitude integration approach (References 13, 33 and 35). The high speed portion of the algorithm is designed to account for high frequency angular and linear oscillations that can rectify into systematic velocity build-up (traditionally denoted as "sculling"); the low speed portion of the algorithm performs the acceleration transformation based on inputs from the high speed algorithm. In general, however, the acceleration-transformation/velocity-integration algorithms have lacked the analytical sophistication of the attitude integration algorithms, being typically limited to first order accuracy under maneuvering conditions.

Virtually no specialized work has been reported for the inertial navigation position integration function. From the writer's understanding, modern day strapdown inertial navigation systems typically generate position as a simple trapezoidal integration of velocity at an update rate equal to or lower than the velocity update frequency. For applications requiring precise position change data in a dynamic environment (e.g., synthetic aperture radar motion compensation), such a rudimentary approach to position integration may prove inadequate. For high resolution applications, the position integration algorithm can also be structured using a two-speed format that is directly analogous to the two-speed attitude/velocity integration algorithm approach. The rectified dynamic angular-rate/acceleration effect measured by the high speed position algorithm has been coined “scrolling” (by the writer) analogous to “coning” and “sculling” measured by the high speed attitude/velocity algorithms.

Section 7.1 of this chapter develops the classical two-speed attitude integration algorithm using a generic exact equation for the high speed portion, and describes a particular version of the high speed portion to illustrate the design of one of the classical high speed second order coning computation algorithms. Section 7.2 utilizes the Section 7.1 attitude algorithm formulation as a model to formulate two-speed acceleration-transformation/velocity-integration algorithms. Section 7.3 then uses Section 7.2 as a framework for the development of position updating algorithms in two forms; a traditional form based on trapezoidal integration and a two-speed high resolution version that includes the scrolling portion. Section 7.4 provides a general discussion of the process followed in selecting algorithms for a particular application and establishing their execution rates. A tabular reference summary of the attitude/velocity/position integration algorithms is presented in Section 7.5.

It should also be noted that Chapter 19 (Section 19.1) describes a new unified two-speed strapdown position/velocity algorithm approach that was originated by the author following the original publication of this book in 2000. The velocity/position translation vectors utilized in the unified approach are calculated at high speed and provide velocity/position change increments to the low speed navigation algorithms of this section.

In retrospect, it is important to recognize that while the original intent of the two-speed approach was to overcome throughput limitations of early computer technology (1965-1975), that limitation is rapidly becoming insignificant with continuing rapid advances in modern high-speed computer technology. This provides the motivation to return to a simpler single speed algorithm structure in which all computations are executed at a repetition rate that is sufficiently high to accurately account for multi-axis high frequency angular rate and acceleration rectification effects. The two-speed algorithms described in this chapter are compatible with compression into such a single speed format as explained in the particular sections in which the algorithms are formulated.

The principal coordinate frames utilized in this chapter are the B, L, N, E and I Frames defined in Section 2.2. Specialized versions of these frames are also defined separately in the particular sections that they are applied.

7.1 ATTITUDE UPDATE ALGORITHMS

Algorithms for the attitude integration process are developed in Section 7.1.1 for direction cosine attitude parameters and in Section 7.1.2 for attitude quaternion parameters.

7.1.1 ATTITUDE DIRECTION COSINE MATRIX (B TO L) UPDATE ALGORITHMS

The digital updating algorithm for the C_B^L direction cosine matrix should ideally achieve the same numerical result at the attitude update times as would the formal continuous integration of the Equations (4.1-1) $\dot{C}_B^L$ expression at the same time instant. In this sense, the numerical history of a continuous integral of (4.1-1) can be viewed as the design requirement for the equivalent digital C_B^L updating algorithm under the same input angular rate profile ($\underline{\omega}_{IB}^B$ and $\underline{\omega}_{IL}^L$ in Equation (4.1-1)). The C_B^L algorithm is constructed by envisioning the body (B) and local level (L) Frame attitude histories in the digital updating world as being constructed of successive discrete attitudes relative to non-rotating inertial space (I) at each update time instant. To be completely general, we also allow that C_B^L updating operations for L Frame angular motion may not necessarily occur at the same time instant that C_B^L is updated for B Frame motion (e.g., for a multi-rate digital computation rate loop structure in which C_B^L is updated at a higher rate for B Frame rotation than for L Frame rotation. In the interests of minimizing computer throughput requirements, the software architecture might have L Frame updates occurring 5 -10 times slower than B Frame updates). We adopt special nomenclature to describe the coordinate frame attitude history where:

$B_{I(m)}$ = Discrete attitude of the B Frame in non-rotating inertial space (I) at computer update time t_m.

m = Computer cycle index for B Frame motion updates to C_B^L.

$L_{I(n)}$ = Discrete attitude of the L Frame in non-rotating inertial space (I) at computer update time t_n.

n = Computer cycle index for L Frame motion updates to C_B^L.

With these definitions, the general updating algorithm for C_B^L is constructed as follows using the Equation (3.2.1-5) direction cosine matrix product chain rule:

$$
\begin{aligned}
C_{B_{I(m)}}^{L_{I(n-1)}} &= C_{B_{I(m-1)}}^{L_{I(n-1)}} \, C_{B_{I(m)}}^{B_{I(m-1)}} \\
C_{B_{I(m)}}^{L_{I(n)}} &= C_{L_{I(n-1)}}^{L_{I(n)}} \, C_{B_{I(m)}}^{L_{I(n-1)}}
\end{aligned}
\qquad (7.1.1\text{-}1)
$$

where

$C_{B_{I(m-1)}}^{L_{I(n-1)}}$ = C_B^L relating the B Frame at time t_{m-1} to the L Frame at time t_{n-1}.

$C_{B_{I(m)}}^{L_{I(n)}}$ = C_B^L relating the B Frame at time t_m to the L Frame at time t_n.

$C_{B_{I(m)}}^{B_{I(m-1)}}$ = Direction cosine matrix that accounts for B Frame rotation relative to inertial space from its attitude at time t_{m-1} to its attitude at time t_m.

$C_{L_{I(n-1)}}^{L_{I(n)}}$ = Direction cosine matrix that accounts for L Frame rotation relative to inertial space from its attitude at time t_{n-1} to its attitude at time t_n.

The algorithm described by Equations (7.1.1-1) relates body (B) and local-level (L) frame attitudes at separate times and provides for B and L Frame inertial angular motion updates to C_B^L at different update rates. Unlike the B Frame (which can be rotating dynamically at 200 - 300 degrees per second), the inertial angular rotation rate of the local level L Frame is generally small and equal to earth's rotation rate plus L Frame angular rate relative to the earth ("transport rate" which is typically never larger than a few earth rates). Consequently, the L Frame update can generally be performed at a lower rate than the B Frame update with comparable accuracy. The B and L Frame motion updates to C_B^L are performed by the $C_{B_{I(m)}}^{B_{I(m-1)}}$ and $C_{L_{I(n-1)}}^{L_{I(n)}}$ terms in Equation (7.1.1-1), algorithms for which are derived in the following subsections.

7.1.1.1 BODY (B) FRAME ROTATION UPDATE

The first expression in Equations (7.1.1-1) updates the C_B^L attitude direction cosine matrix using $C_{B_{I(m)}}^{B_{I(m-1)}}$ as follows to account for angular rotation of the strapdown sensor (body) B Frame relative to non-rotating space $\underline{\omega}_{IB}^B$:

$$C_{B_{I(m)}}^{L_{I(n-1)}} = C_{B_{I(m-1)}}^{L_{I(n-1)}} \, C_{B_{I(m)}}^{B_{I(m-1)}} \qquad (7.1.1.1\text{-}1)$$

with, formally:

$$C_{B_{I(m)}}^{B_{I(m-1)}} = I + \int_{t_{m-1}}^{t_m} \dot{C}_{B(t)}^{B_{I(m-1)}} \, dt \qquad (7.1.1.1\text{-}2)$$

and where

I = Identity matrix.

$C_{B(t)}^{B_{I(m-1)}}$ = Direction cosine matrix relating the B Frame attitude at an arbitrary time in the interval t_{m-1} to t_m, to its $B_{I(m-1)}$ attitude.

The $C_{B_{I(m)}}^{B_{I(m-1)}}$ matrix can also be expressed in terms of a rotation vector defining the Frame $B_{I(m)}$ attitude relative to Frame $B_{I(m-1)}$. Applying Equations (3.2.2.1-8) and (3.2.2.1-9) obtains:

$$C_{B_{I(m)}}^{B_{I(m-1)}} = I + \frac{\sin \phi_m}{\phi_m}\left(\underline{\phi}_m \times\right) + \frac{(1 - \cos \phi_m)}{\phi_m^2}\left(\underline{\phi}_m \times\right)\left(\underline{\phi}_m \times\right) \tag{7.1.1.1-3}$$

$$\frac{\sin \phi_m}{\phi_m} = 1 - \frac{\phi_m^{\,2}}{3\,!} + \frac{\phi_m^{\,4}}{5\,!} - \cdots \qquad \frac{(1 - \cos \phi_m)}{\phi_m^{\,2}} = \frac{1}{2\,!} - \frac{\phi_m^{\,2}}{4\,!} + \frac{\phi_m^{\,4}}{6\,!} - \cdots$$

where

$\underline{\phi}_m$ = Rotation vector defining the Frame $B_{I(m)}$ attitude relative to Frame $B_{I(m-1)}$ at time t_m.

ϕ_m = Magnitude of $\underline{\phi}_m$.

The $\underline{\phi}_m$ rotation vector can be computed by treating $\underline{\phi}$ as a general rotation vector defining the general B Frame attitude relative to Frame $B_{I(m-1)}$ for time greater than t_{m-1}. Then $\underline{\phi}$ is calculated as the integral from time t_{m-1} of the general $\dot{\underline{\phi}}$ equation, with $\underline{\phi}$ for Equation (7.1.1.1-3) evaluated as the integral solution at time t_m. Treating Frame $B_{I(m-1)}$ for $\underline{\phi}$ definition as the non-rotating inertial reference frame (I), we obtain the following for the general $\dot{\underline{\phi}}$ expression by application of Equation (3.3.5-14) with general Frame A replaced by inertial Frame I for angular rate description:

$$\dot{\underline{\phi}} = \underline{\omega}_{IB}^{B} + \frac{1}{2}\,\underline{\phi} \times \underline{\omega}_{IB}^{B} + \frac{1}{\phi^2}\left(1 - \frac{\phi \sin \phi}{2\left(1 - \cos \phi\right)}\right)\underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}_{IB}^{B}\right) \tag{7.1.1.1-4}$$

where

$\underline{\phi}$ = Rotation vector defining the general attitude of Frame B relative to Frame $B_{I(m-1)}$ for time greater than t_{m-1}.

$\underline{\omega}_{IB}^{B}$ = Angular rotation rate of Frame B relative to inertial space as described in Frame B axes.

Equation (7.1.1.1-4), commonly referred to as the Bortz equation (See Section 7.0), relates the change in body B Frame attitude to the B Frame angular rate (as would be measured by strapdown angular rate sensors). The attitude rotation vector $\underline{\phi}_m$ for Equation (7.1.1.1-3) is then obtained as the integral of (7.1.1.1-4) from time t_{m-1}, evaluated at time t_m:

$$\underline{\phi}(t) = \int_{t_{m-1}}^{t} \dot{\underline{\phi}}(\tau)\, d\tau \qquad \underline{\phi}_m = \underline{\phi}(t_m) \tag{7.1.1.1-5}$$

where

τ = Running integration time variable.

To reduce the number of computations involved in calculating $\dot{\underline{\phi}}$ with Equation (7.1.1.1-4), simplifying assumptions are typically incorporated. For example, through a power series expansion, the scalar multiplier of the $\underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}_{IB}^{B}\right)$ term in (7.1.1.1-4) can be approximated as:

$$\frac{1}{\phi^2}\left(1 - \frac{\phi \sin \phi}{2\,(1 - \cos \phi)}\right) = \frac{1}{12}\left(1 + \frac{1}{60}\,\phi^2 + \cdots\right) \approx \frac{1}{12} \tag{7.1.1.1-6}$$

hence, Equation (7.1.1.1-4)) to second order in $\underline{\phi}$ is given by

$$\dot{\underline{\phi}} \approx \underline{\omega}_{IB}^{B} + \frac{1}{2}\,\underline{\phi} \times \underline{\omega}_{IB}^{B} + \frac{1}{12}\,\underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}_{IB}^{B}\right) \tag{7.1.1.1-7}$$

Through simulation and analysis (analytical expansion under hypothesized analytically definable angular motion conditions) it can be shown that to second order accuracy in $\underline{\phi}$:

$$\frac{1}{2}\left(\underline{\phi} \times \underline{\omega}_{IB}^{B}\right) + \frac{1}{12}\,\underline{\phi} \times \left(\underline{\phi} \times \underline{\omega}_{IB}^{B}\right) \approx \frac{1}{2}\,\underline{\alpha} \times \underline{\omega}_{IB}^{B} \tag{7.1.1.1-8}$$

with

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}^{B}\, d\tau \tag{7.1.1.1-9}$$

where

$\underline{\alpha}(t)$ = Integral of $\underline{\omega}_{IB}^{B}$ from time t_{m-1} to time t.

Equation (7.1.1.1-8) is extremely significant because it enables Equation (7.1.1.1-4) to be simplified to second order accuracy (i.e., in error to third order in $\underline{\phi}$) by retaining only first order terms. Thus, Equation (7.1.1.1-4) becomes to <u>second</u> order accuracy:

$$\underline{\dot{\phi}} \approx \underline{\omega}_{IB}^{B} + \frac{1}{2}\,\underline{\alpha} \times \underline{\omega}_{IB}^{B} \tag{7.1.1.1-10}$$

With (7.1.1.1-10), Equation (7.1.1.1-5) is given by:

$$\underline{\phi}_m = \int_{t_{m-1}}^{t_m} \left[\underline{\omega}_{IB}^{B} + \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\omega}_{IB}^{B} \right) \right] dt \tag{7.1.1.1-11}$$

Finally, using Equation (7.1.1.1-9) we obtain:

$$\underline{\phi}_m = \underline{\alpha}_m + \underline{\beta}_m \tag{7.1.1.1-12}$$

with

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}^{B}\, d\tau \qquad \underline{\alpha}_m = \underline{\alpha}(t_m)$$
$$\underline{\beta}_m = \frac{1}{2}\int_{t_{m-1}}^{t_m} \left(\underline{\alpha}(t) \times \underline{\omega}_{IB}^{B} \right) dt \tag{7.1.1.1-13}$$

where

$\underline{\beta}_m$ = Coning attitude motion from t_{m-1} to t_m .

The $\underline{\beta}_m$ term has been coined the "coning" term because it measures the effect of "coning motion" components present in $\underline{\omega}_{IB}^{B}$. "Coning motion" is defined as the condition when an angular rate vector is itself rotating. For $\underline{\omega}_{IB}^{B}$ exhibiting pure coning motion (the $\underline{\omega}_{IB}^{B}$ magnitude being constant but the vector rotating) a fixed axis in the B Frame that is approximately perpendicular to the plane of the rotating $\underline{\omega}_{IB}^{B}$ vector will generate a conical surface as the angular rate motion ensues (hence, the term "coning" to describe the motion). Under coning angular motion conditions, B Frame axes perpendicular to $\underline{\omega}_{IB}^{B}$ appear to oscillate (in contrast with non-coning or "spinning" angular motion in which axes perpendicular to $\underline{\omega}_{IB}^{B}$ rotate around $\underline{\omega}_{IB}^{B}$).

For situations when $\underline{\omega}_{IB}^{B}$ is not rotating (i.e., parallel to a stationary non-rotating line) it is easily seen from Equation (7.1.1.1-13) that $\underline{\alpha}(t)$ will be parallel to $\underline{\omega}_{IB}^{B}$, hence, the cross-product in the $\underline{\beta}_m$ integrand will be zero and $\underline{\beta}_m$ will be zero. Under these conditions, Equation (7.1.1.1-11) reduces to the simplified form:

$$\underline{\phi}_m = \int_{t_{m-1}}^{t_m} \underline{\omega}_{IB}^{B} \, dt \qquad \text{When } \underline{\omega}_{IB}^{B} \text{ Is Not Rotating} \tag{7.1.1.1-14}$$

It should be noted that Equation (7.1.1.1-14) also applies to the exact $\underline{\phi}_m$ Equations (7.1.1.1-4) - (7.1.1.1-5) (i.e., without approximation). This is readily verified by observing from Equation (7.1.1.1-4) that $\underline{\phi}(t)$ will initially be aligned with $\underline{\omega}_{IB}^{B}$ as the $\dot{\underline{\phi}}(t)$ integration begins, and will then remain parallel to $\underline{\omega}_{IB}^{B}$ because its cross-products with $\underline{\omega}_{IB}^{B}$ in the $\dot{\underline{\phi}}(t)$ expression will remain zero. Under these conditions, Equations (7.1.1.1-4) - (7.1.1.1-5) also reduce to (7.1.1.1-14).

7.1.1.1.1 Integrated Rate And Coning Computation Algorithms

A discrete digital algorithm form of the $\underline{\alpha}_m$ integrated rate and $\underline{\beta}_m$ coning expressions in Equations (7.1.1.1-13) can be developed by considering $\underline{\beta}_m$ to be the value at $t = t_m$ of the general function $\underline{\beta}(t)$. From (7.1.1.1-13):

$$\underline{\beta}(t) = \frac{1}{2} \int_{t_{m-1}}^{t} \left(\underline{\alpha}(\tau) \times \underline{\omega}_{IB}^{B} \right) d\tau \tag{7.1.1.1.1-1}$$

Let us consider the integration of (7.1.1.1.1-1) as divided into portions up to and after a general time t_{l-1} within the t_{m-1} to t_m interval so that (7.1.1.1.1-1) is equivalently:

$$\underline{\beta}(t) = \underline{\beta}_{l-1} + \Delta\underline{\beta}(t) \qquad \Delta\underline{\beta}(t) = \frac{1}{2} \int_{t_{l-1}}^{t} \left(\underline{\alpha}(\tau) \times \underline{\omega}_{IB}^{B} \right) d\tau \tag{7.1.1.1.1-2}$$

with

$$\underline{\beta}_m = \underline{\beta}(t_m) \tag{7.1.1.1.1-3}$$

where

$\underline{\beta}_{l-1}$ = Value of $\underline{\beta}(t)$ at $t = t_{l-1}$.

l = Computer cycle index for $t = t_l$ cycle times. Note that by its definition, the l cycle index is faster than the m cycle index.

We now define the next l cycle time point t_l to be within the t_{m-1} to t_m interval so that at t_l Equations (7.1.1.1.1-2) and (7.1.1.1.1-3), including initial conditions, become:

$$\underline{\beta}_l = \underline{\beta}_{l-1} + \Delta\underline{\beta}_l \qquad \underline{\beta}_m = \underline{\beta}_l(t_l = t_m) \qquad \underline{\beta}_l = 0 \quad \text{At } t = t_{m-1}$$
$$\Delta\underline{\beta}_l = \frac{1}{2}\int_{t_{l-1}}^{t_l} \left(\underline{\alpha}(t) \times \underline{\omega}_{IB}^B\right) dt \tag{7.1.1.1.1-4}$$

Through a similar process, the $\underline{\alpha}(t)$ expression for (7.1.1.1.1-4) (and $\underline{\alpha}_m$ for (7.1.1.1-12)) is obtained by manipulation of $\underline{\alpha}(t)$ in Equations (7.1.1.1-13):

$$\underline{\alpha}(t) = \underline{\alpha}_{l-1} + \Delta\underline{\alpha}(t)$$
$$\Delta\underline{\alpha}(t) = \int_{t_{l-1}}^{t} \underline{\omega}_{IB}^B \, d\tau \qquad \Delta\underline{\alpha}_l = \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}^B \, dt \tag{7.1.1.1.1-5}$$
$$\underline{\alpha}_l = \underline{\alpha}_{l-1} + \Delta\underline{\alpha}_l \qquad \underline{\alpha}_m = \underline{\alpha}_l(t_l = t_m) \qquad \underline{\alpha}_l = 0 \quad \text{At } t = t_{m-1}.$$

Equations (7.1.1.1.1-4) and (7.1.1.1.1-5) constitute the construct of a digital recursive algorithm at the *l* computer cycle rate for calculating the $\underline{\beta}_m$ coning term as a summation of changes in $\underline{\beta}$ over the t_{m-1} to t_m interval. It remains to determine a digital equivalent for the $\Delta\underline{\beta}_l$ integral term in (7.1.1.1.1-4).

We begin by substituting $\underline{\alpha}(t)$ from (7.1.1.1.1-5) into (7.1.1.1.1-4) for $\Delta\underline{\beta}_l$ and incorporate the definition of $\Delta\underline{\alpha}_l$ from (7.1.1.1.1-5). The result is:

$$\Delta\underline{\beta}_l = \frac{1}{2}\left(\underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l\right) + \frac{1}{2}\int_{t_{l-1}}^{t_l} \left(\Delta\underline{\alpha}(t) \times \underline{\omega}_{IB}^B\right) dt \tag{7.1.1.1.1-6}$$

Continuing work in attitude algorithm development has centered on the design of digital algorithms for evaluating the $\Delta\underline{\beta}_l$ integral term in coning Equation (7.1.1.1.1-6). In general, the methods utilized assume a general analytical form for the angular rate profile $\underline{\omega}_{IB}^B$ in the t_{l-1} to t_l time interval (e.g., a truncated general polynomial in time). The (7.1.1.1.1-6) integral is then analytically determined as a function of the general rate profile coefficients (e.g., the polynomial coefficients). Finally, the coefficients for the angular rate profile are calculated to fit successive integrated angular rate increment measurements. For the example that follows, the angular rate profile is approximated as a constant plus a linear build-up in time with the constant and ramping coefficients calculated from the current and previous values of $\Delta\underline{\alpha}_l$. A more sophisticated version of this algorithm might include a parabolic-with-time term in the assumed angular rate profile, utilizing the current, past, and past-past values of $\Delta\underline{\alpha}_l$ for coefficient determination. Recent work in this area (References 11 and 12), calculates the angular rate profile coefficients from angular rate sensor measurements taken within the t_{l-1} to t_l time interval, thereby incorporating a third computation cycle rate into the overall attitude update process architecture: attitude (C_B^L) update, coning ($\underline{\beta}_l$) update (as discussed thus far), and sensor sampling for the coning update. Refinements on this technique (References 11 and 12) utilize a general angular rate profile that is defined directly in terms of its impact on the Equation (7.1.1.1.1-6) integral as a sum of weighted cross-products between successive integrated angular rate increment sensor samples taken during the t_{l-1} to t_l time interval (similar to the approach presented in Reference 22 over the t_{m-1} to t_m interval). The weighting coefficients are then "optimized" for best average performance in a pure coning environment (i.e., $\underline{\omega}_{IB}^B$ constant in magnitude but rotating). Each of the design approaches is based on curve fitting techniques for an assumed angular rate profile shape. Each resulting algorithm behaves differently in rate environments for which it was not designed and in the presence of angular rate sensor quantization noise. Selection of the "preferred" algorithm should include simulation analysis to confirm acceptable performance under operational rate environments and sensor noise characteristics.

We conclude this section by providing an example of an algorithm for the Equation (7.1.1.1.1-6) integral term based on the body rate term $\underline{\omega}_{IB}^B$ being approximated to first order by the truncated power series expansion:

$$\underline{\omega}_{IB}^B \approx \underline{A} + \underline{B}\left(t - t_{l-1}\right) \tag{7.1.1.1.1-7}$$

where

$\underline{A}$ and $\underline{B}$ = Constants.

With (7.1.1.1.1-7) the $\Delta\underline{\alpha}(t)$ expression in (7.1.1.1.1-5) becomes:

$$\Delta\underline{\alpha}(t) = \int_{t_{l-1}}^{t} \underline{\omega}_{IB}^B \, d\tau = \underline{A}\left(t - t_{l-1}\right) + \frac{1}{2}\,\underline{B}\left(t - t_{l-1}\right)^2 \tag{7.1.1.1.1-8}$$

Substituting (7.1.1.1.1-8) into the integral term in $\Delta\underline{\beta}_l$ Equation (7.1.1.1.1-6) then yields:

$$\frac{1}{2}\int_{t_{l-1}}^{t_l}\left(\underline{\Delta\alpha}(t)\times\underline{\omega}_{IB}^{B}\right)dt = \frac{1}{2}\int_{t_{l-1}}^{t_l}\left[\underline{A}\left(t - t_{l-1}\right)+\frac{1}{2}\,\underline{B}\left(t - t_{l-1}\right)^2\right]\times\left[\underline{A}+\underline{B}\left(t - t_{l-1}\right)\right]dt$$

$$= \frac{1}{2}\int_{t_{l-1}}^{t_l}\left[\left(\underline{A}\times\underline{A}\right)\left(t - t_{l-1}\right)+\left(\underline{A}\times\underline{B}\right)\left(t - t_{l-1}\right)^2 + \frac{1}{2}\left(\underline{B}\times\underline{A}\right)\left(t - t_{l-1}\right)^2+\frac{1}{2}\left(\underline{B}\times\underline{B}\right)\left(t - t_{l-1}\right)^3\right]dt \qquad (7.1.1.1.1\text{-}9)$$

Noting that $\underline{A}\times\underline{A} = 0$, $\underline{B}\times\underline{B} = 0$ and $\underline{A}\times\underline{B} = -\,\underline{B}\times\underline{A}$ allows (7.1.1.1.1-9) to simplify to:

$$\frac{1}{2}\int_{t_{l-1}}^{t_l}\left(\underline{\Delta\alpha}(t)\ \times\underline{\omega}_{IB}^{B}\right)dt = \frac{1}{2}\int_{t_{l-1}}^{t_l}\left[\left(\underline{A}\times\underline{B}\right)\left(t - t_{l-1}\right)^2 - \frac{1}{2}\left(\underline{A}\times\underline{B}\right)\left(t - t_{l-1}\right)^2\right]dt$$

$$= \frac{1}{4}\int_{t_{l-1}}^{t_l}\left(\underline{A}\times\underline{B}\right)\left(t - t_{l-1}\right)^2 dt = \frac{1}{12}\left(\underline{A}\times\underline{B}\right)T_l^3 \qquad (7.1.1.1.1\text{-}10)$$

where

T_l = Time interval $t_l - t_{l-1}$ (i.e., the l cycle computation period).

The $\underline{A}$ and $\underline{B}$ terms in (7.1.1.1.1-10) can be expressed as functions of past and current $\underline{\Delta\alpha}$ values as follows. From (7.1.1.1.1-8) we have:

$$\underline{\Delta\alpha}_l = \int_{t_{l-1}}^{t_l}\underline{\omega}_{IB}^{B}\,dt = \underline{A}\left(t_l - t_{l-1}\right)+\frac{1}{2}\,\underline{B}\left(t_l - t_{l-1}\right)^2 = \underline{A}\,T_l + \frac{1}{2}\,\underline{B}\,T_l^2 \qquad (7.1.1.1.1\text{-}11)$$

and for the previous cycle, still using (7.1.1.1.1-7) for $\underline{\omega}_{IB}^{B}$:

$$\underline{\Delta\alpha}_{l-1} = \int_{t_{l-2}}^{t_{l-1}}\underline{\omega}_{IB}^{B}\,dt = \underline{A}\left(t_{l-1} - t_{l-2}\right)-\frac{1}{2}\,\underline{B}\left(t_{l-2} - t_{l-1}\right)^2 = \underline{A}\,T_l - \frac{1}{2}\,\underline{B}\,T_l^2 \qquad (7.1.1.1.1\text{-}12)$$

in which it has been assumed that the computer l cycle period is fixed so that $t_l - t_{l-1}$ also equals $t_{l-1} - t_{l-2}$.

Equations (7.1.1.1.1-11) and (7.1.1.1.1-12) when added and differenced yields the following:

$$\underline{\Delta\alpha}_l + \underline{\Delta\alpha}_{l-1} = 2\,\underline{A}\,T_l \qquad \underline{\Delta\alpha}_l - \underline{\Delta\alpha}_{l-1} = \underline{B}\,T_l^2 \qquad (7.1.1.1.1\text{-}13)$$

Rearrangement of Equations (7.1.1.1.1-13) shows that:

$$\underline{A} = \frac{1}{2\,T_l}\left(\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) \qquad \underline{B} = \frac{1}{T_l^2}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right) \tag{7.1.1.1.1-14}$$

Substituting Equation (7.1.1.1.1-14) into (7.1.1.1.1-10) and noting that $\Delta\underline{\alpha}_l \times \Delta\underline{\alpha}_l = 0$ and $\Delta\underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_{l-1} = 0$ gives:

$$\begin{aligned}\frac{1}{2}\int_{t_{l-1}}^{t_l}\left(\Delta\underline{\alpha}(t) \times \underline{\omega}_{IB}^{B}\right) dt &= \frac{1}{12}\left[\frac{1}{2\,T_l}\left(\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) \times \frac{1}{T_l^2}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] T_l^3 \\ &= \frac{1}{12}\left[\left(\frac{1}{2}\right)\left(-\,\Delta\underline{\alpha}_l \times \Delta\underline{\alpha}_{l-1} + \Delta\underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l\right)\right] = \frac{1}{12}\left(\Delta\underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l\right)\end{aligned} \tag{7.1.1.1.1-15}$$

Substituting (7.1.1.1.1-15) into (7.1.1.1.1-6) then finally obtains:

$$\Delta\underline{\beta}_l = \frac{1}{2}\left(\underline{\alpha}_{l-1} + \frac{1}{6}\,\Delta\underline{\alpha}_{l-1}\right) \times \Delta\underline{\alpha}_l \tag{7.1.1.1.1-16}$$

The overall digital algorithm for $\underline{\alpha}_m$ and $\underline{\beta}_m$ in Equation (7.1.1.1-12) is determined from the above results as a composite of Equations (7.1.1.1.1-4), (7.1.1.1.1-5) and (7.1.1.1.1-16):

$$\Delta\underline{\alpha}_l = \int_{t_{l-1}}^{t_l} d\underline{\alpha} \qquad \text{Summation Of Integrated Angular Rate Output Increments From Angular Rate Sensors.}$$

$$\underline{\alpha}_l = \underline{\alpha}_{l-1} + \Delta\underline{\alpha}_l \qquad \underline{\alpha}_m = \underline{\alpha}_l(t_l = t_m) \qquad \underline{\alpha}_l = 0 \quad \text{At } t = t_{m-1} \tag{7.1.1.1.1-17}$$

$$\Delta\underline{\beta}_l = \frac{1}{2}\left(\underline{\alpha}_{l-1} + \frac{1}{6}\,\Delta\underline{\alpha}_{l-1}\right) \times \Delta\underline{\alpha}_l$$

$$\underline{\beta}_l = \underline{\beta}_{l-1} + \Delta\underline{\beta}_l \qquad \underline{\beta}_m = \underline{\beta}_l(t_l = t_m) \qquad \underline{\beta}_l = 0 \quad \text{At } t = t_{m-1} \tag{7.1.1.1.1-18}$$

where

$d\underline{\alpha} = \underline{\omega}_{IB}^{B}\,dt$ = Differential integrated angular rate increment (i.e., analytical representation of pulse output from strapdown angular rate sensors).

Equation (7.1.1.1.1-18) has been classified as a "second order algorithm" for $\underline{\beta}_m$ because it includes current and past cycle $\Delta\underline{\alpha}$ products in the $\Delta\underline{\beta}_l$ equation. From the analysis leading to Equation (7.1.1.1.1-15), the l and l-1 $\Delta\underline{\alpha}$ product term in $\Delta\underline{\beta}_l$ (i.e., the 1 / 6 term) stems from the approximation of linearly ramping angular rate in the t_{l-2} to t_l time interval. If the angular rate was approximated as a parabolically varying function of time, a "third order algorithm" would result containing l, l-1, and l-2 $\Delta\underline{\alpha}$ products. If the angular rate was approximated as a constant over t_{l-1} to t_l, the 1 / 6 term for $\Delta\underline{\beta}_l$ in (7.1.1.1.1-18) would vanish, resulting in a "first order algorithm" for $\underline{\beta}_m$. Finally, if angular rates are slowly varying we can approximate $\underline{\beta}_m$ as being equal to zero. Alternatively (and more accurately), we can set the l cycle rate equal to the m cycle rate which equates $\underline{\beta}_m$ in Equations (7.1.1.1.1-18) to $\Delta\underline{\beta}_l$ calculated once at time t_m (and noting from the initial condition definition in (7.1.1.1.1-17) that $\underline{\alpha}_{l-1}$ would be zero). Note, that setting the l and m rates equal can also be achieved by increasing the m rate to match the l rate. The result would be a single high speed higher order algorithm with a simpler software architecture than the two-speed approach, but requiring more throughput. Continuing advances in the speed of modern day computers may make this the preferred approach for the future.

7.1.1.2 LOCAL LEVEL (L) FRAME ROTATION UPDATE

The remaining part of the C_B^L attitude direction cosine matrix update accounts for the $\underline{\omega}_{IL}^L$ rotation rate of the local-level coordinate L Frame relative to non-rotating inertial space. The L Frame update is given by Equation (7.1.1-1) as follows:

$$C_{B_{I(m)}}^{L_{I(n)}} = C_{L_{I(n-1)}}^{L_{I(n)}} \, C_{B_{I(m)}}^{L_{I(n-1)}} \qquad (7.1.1.2\text{-}1)$$

The $C_{L_{I(n-1)}}^{L_{I(n)}}$ term in Equation (7.1.1.2-1) relates the local-level L Frame at time t_{n-1} to the L Frame at time t_n. A derivation that directly parallels that used to determine $C_{B_{I(m)}}^{B_{I(m-1)}}$ in Section 7.1.1.1 follows for $C_{L_{I(n-1)}}^{L_{I(n)}}$.

The formal definition for $C_{L_{I(n-1)}}^{L_{I(n)}}$ is:

$$C_{L_{I(n-1)}}^{L_{I(n)}} = I + \int_{t_{n-1}}^{t_n} \dot{C}_{L_{I(n-1)}}^{L(t)} \, dt \qquad (7.1.1.2\text{-}2)$$

where

$C_{L_{I(n-1)}}^{L(t)}$ = Direction cosine matrix relating the L Frame attitude at an arbitrary time t in the interval t_{n-1} to t_n, to its $L_{I(n-1)}$ attitude.

The $C_{L_{I(n-1)}}^{L_{I(n)}}$ matrix can also be expressed in terms of the rotation vector defining the Frame $L_{I(n)}$ attitude relative to Frame $L_{I(n-1)}$. Applying Equations (3.2.2.1-8) and (3.2.2.1-9) obtains:

$$C_{L_{I(n-1)}}^{L_{I(n)}} = I - \frac{\sin \zeta_n}{\zeta_n}\left(\underline{\zeta}_n \times\right) + \frac{(1 - \cos \zeta_n)}{\zeta_n^{\,2}}\left(\underline{\zeta}_n \times\right)\left(\underline{\zeta}_n \times\right)$$

$$\frac{\sin \zeta_n}{\zeta_n} = 1 - \frac{\zeta_n^{\,2}}{3\,!} + \frac{\zeta_n^{\,4}}{5\,!} - \cdots \qquad \frac{(1 - \cos \zeta_n)}{\zeta_n^{\,2}} = \frac{1}{2\,!} - \frac{\zeta_n^{\,2}}{4\,!} + \frac{\zeta_n^{\,4}}{6\,!} - \cdots \tag{7.1.1.2-3}$$

where

$\underline{\zeta}_n$ = Rotation vector defining the Frame $L_{I(n)}$ attitude at time t_n relative to Frame $L_{I(n-1)}$ attitude at time t_{n-1}.

ζ_n = Magnitude of $\underline{\zeta}_n$.

Note in Equation (7.1.1.2-3) that the sign for the $\frac{\sin \zeta_n}{\zeta_n}\left(\underline{\zeta}_n \times\right)$ term is negative in contrast with the similar term in the Equation (7.1.1.1-3) $C_{B_{I(m)}}^{B_{I(m-1)}}$ expression. This is because the $C_{L_{I(n-1)}}^{L_{I(n)}}$ matrix has the opposite phase sense from $C_{B_{I(m)}}^{B_{I(m-1)}}$ (or C_B^A in (3.2.2.1-8)) in that $C_{L_{I(n-1)}}^{L_{I(n)}}$ transforms vectors from $L_{I(n-1)}$ to $L_{I(n)}$ while $C_{B_{I(m)}}^{B_{I(m-1)}}$ transforms vectors from $B_{I(m)}$ to $B_{I(m-1)}$. As such, the $C_{L_{I(n-1)}}^{L_{I(n)}}$ form in Equation (7.1.1.2-3) is the transpose of the Equation (7.1.1.1-3) $C_{B_{I(m)}}^{B_{I(m-1)}}$ expression form.

Because the t_{n-1} to t_n update cycle is relatively short, $\underline{\zeta}_n$ will be very small in magnitude. Since $\underline{\omega}_{IL}^{L}$ is small and slowly changing over a typical t_{n-1} to t_n update cycle (due to small changes in velocity and position over this time period) the L frame rate vector $\underline{\omega}_{IL}^{L}$ can be approximated as non-rotating. The result is that $\underline{\zeta}_n$ for (7.1.1.2-3) can be calculated as the integral of the simplified form of the Equation (3.3.5-14) rotation vector rate equation in which the cross-product terms are neglected:

$$\underline{\zeta}_n \approx \int_{t_{n-1}}^{t_n} \underline{\omega}_{IL}^{L}\, dt \tag{7.1.1.2-4}$$

We note in passing that based on the smallness of $\underline{\zeta}_n$ as discussed above, Equations (7.1.1.2-3) for $C_{L_{I(n-1)}}^{L_{I(n)}}$ can also be simplified. For example, a second order version (accurate to second order in $\underline{\zeta}_n$) is from (7.1.1.2-3):

$$C_{L_{I(n-1)}}^{L_{I(n)}} \approx I - \left(\underline{\zeta}_n \times\right) + \frac{1}{2}\left(\underline{\zeta}_n \times\right)\left(\underline{\zeta}_n \times\right) \tag{7.1.1.2-5}$$

The computer memory/throughput advantages of utilizing a simplified form of (7.1.1.2-3) for $C_{L_{I(n-1)}}^{L_{I(n)}}$ (such as (7.1.1.2-5)) are trivial for today's modern computer technology compared to the disadvantages of increased software validation/documentation complexity and loss in accuracy. The accuracy loss is generally minor during navigation, however, it might not be negligible during initial alignment operations when the $C_{L_{I(n-1)}}^{L_{I(n)}}$ matrix is used to apply tilt updates to C_B^L (e.g., see Section 6.1.2). Initial tilt corrections to C_B^L can be fairly large (e.g., 0.1 to 1.0 deg) which can produce undesirable errors in C_B^L during the initial alignment process if too simplified a version of (7.1.1.2-3) is utilized. The closed-loop servo action of the Section 6.1.2 initial alignment operations would eventually correct the resulting attitude error generated in C_B^L, however, it could leave a residual orthogonality/normality error in the C_B^L rows (and columns). The result would be the requirement to include an orthogonality/normalization correction algorithm (See Section 7.1.1.3) as an outer loop in the C_B^L update processing.

7.1.1.2.1 Integrated Rate Algorithm

A discrete digital algorithm for the Equation (7.1.1.2-4) $\underline{\zeta}_n$ integral can be constructed by first combining Equations (4.1.1-1) and (4.1.1-6) to obtain for the integrand:

$$\underline{\omega}_{IL}^{L} = C_N^L \left[\underline{\omega}_{IE}^{N} + \rho_{ZN}\, \underline{u}_{ZN}^{N} + F_C^N \left(\underline{u}_{ZN}^{N} \times \underline{v}^N\right)\right] \tag{7.1.1.2.1-1}$$

and then approximating:

$$\underline{\omega}_{IL}^{L} \approx C_N^L \left[\underline{\omega}_{IE_{n-1/2}}^{N} + \rho_{ZN_{n-1/2}}\, \underline{u}_{ZN}^{N} + F_{C_{n-1/2}}^N \left(\underline{u}_{ZN}^{N} \times \underline{v}^N\right)\right] \tag{7.1.1.2.1-2}$$

where

n-1/2 = Subscript indicating value for parameter midway between times t_{n-1} and t_n.

Using (7.1.1.2.1-2) in (7.1.1.2-4) then obtains:

$$\underline{\zeta}_n \approx C_N^L \left[\underline{\omega}_{IE_{n-1/2}}^{N}\, T_n + \rho_{ZN_{n-1/2}}\, \underline{u}_{ZN}^{N}\, T_n + F_{C_{n-1/2}}^N \left(\underline{u}_{ZN}^{N} \times \sum^{j} \Delta\underline{R}_m^N\right)\right] \tag{7.1.1.2.1-3}$$

with $\underline{\omega}_{IE}^{N}$ evaluated using Equation (4.1.1-3) and:

$$\Delta\underline{R}_m^N \equiv \int_{t_{m-1}}^{t_m} \underline{v}^N \, dt \qquad (7.1.1.2.1\text{-}4)$$

where

T_n = Computer n cycle update period $t_n - t_{n-1}$.

j = Number of computer m cycles over the t_{n-1} to t_n n cycle computer update period.

The $(\,)_{n-1/2}$ terms in (7.1.1.2.1-3) are all functions of position, which is updated in Section 7.3 at the n cycle rate preceding the attitude update. Hence, current and past n cycle values of the ( ) parameters are available for calculating the $(\,)_{n-1/2}$ terms in (7.1.1.2.1-3). For example, a linear interpolation formula using the current and past computed values for ( ) would be:

$$(\,)_{n-1/2} \approx \frac{1}{2}\left[(\,)_n + (\,)_{n-1}\right] \qquad (7.1.1.2.1\text{-}5)$$

In Section 7.2 we find that the $\underline{v}^N$ velocity update precedes the attitude update. Therefore, current and past m cycle values of $\underline{v}^N$ are available for evaluating the Equation (7.1.1.2.1-4) integral for $\Delta\underline{R}_n^N$. Using a trapezoidal integration algorithm for (7.1.1.2.1-4) obtains:

$$\Delta\underline{R}_m^N \approx \frac{1}{2}\left(\underline{v}_m^N + \underline{v}_{m-1}^N\right) T_m \qquad (7.1.1.2.1\text{-}6)$$

where

T_m = Computer m cycle update period $t_m - t_{m-1}$.

Section 7.3 also develops a high resolution version of $\Delta\underline{R}_m^N$ for precision position updating that accounts for dynamic angular rates and accelerations within the m-1 to m cycle update interval.

7.1.1.3 ATTITUDE DIRECTION COSINE MATRIX (B TO L) NORMALIZATION AND ORTHOGONALIZATION CORRECTIONS

In addition to the basic C_B^L update algorithms given previously, a normalization and orthogonalization algorithm is frequently included to insure that the C_B^L rows remain normal and orthogonal to each other. Factors that cause orthogonality and normalization errors in the C_B^L updating algorithms include orthogonality/normality C_B^L initialization errors, software programming error, round-off error due to insufficient computer word-length for the total number of C_B^L algorithm update cycles expected, and an insufficient number of terms carried in the Equation (7.1.1.1-3) and (7.1.1.2-3) Taylor series expansions. It is important to note (as discussed in Section 3.5.1) that orthogonality and normalization errors can only be produced from errors in the software implementation of Equations (7.1.1-1), (7.1.1.1-3) and (7.1.1.2-3); not from errors in the algorithms feeding these equations or from inertial sensor input errors. The overall C_B^L update software design/verification process must assure error free software programming and that the Taylor series truncation error, algorithm update rates selected, and computer round-off error is acceptable for the angular rate environment anticipated over the expected navigation time period (as discussed in more detail in Chapters 10 and 11). Never-the-less, inclusion of a C_B^L orthogonality/normality correction algorithm has been traditionally employed in most strapdown software packages for enhanced accuracy and to relax the more stringent requirement of not allowing any orthogonality/normalization error in the basic C_B^L updating operations.

The orthogonality/normality correction algorithm for the C_B^L direction cosine matrix is developed in this section based on maintaining the rows of C_B^L unity in magnitude and perpendicular to each other. Orthogonalizing and normalizing the rows of C_B^L also achieves the same result for the C_B^L columns as demonstrated by Equation (3.5.1-18) of Section 3.5.1. Equations (3.5.1-9) and (3.5.1-15) show that the orthogonality/normality error in C_B^L can be calculated from:

$$E_{SYM} = \frac{1}{2}\left[\hat{C}_B^L \left(\hat{C}_B^L\right)^T - I\right] \qquad (7.1.1.3\text{-}1)$$

where

$\hat{\ }$ = Designation for parameter calculated in the INS computer, hence, containing error. The same parameter without the $\hat{\ }$ designation will be considered in this section to be the idealized error free value.

E_{SYM} = Error in the <u>rows</u> of $\hat{C}_B^L$ characterized by symmetry about the $\hat{C}_B^L$ diagonal. E_{SYM} is a symmetric matrix. The E_{SYM} off-diagonal element in row i, column j equals <u>half</u> the perpendicularity error between rows i and j of $\hat{C}_B^L$ (positive for $\hat{C}_B^L$ rows i and j being less than 90 degrees apart). The E_{SYM} i^{th} diagonal element equals the normality error (error in magnitude) of $\hat{C}_B^L$ row i.

Based on the above definitions and generalized Equations (3.5.1-13) and (3.5.1-15), the normality error in row i of $\hat{C}_B^L$ can be expressed as:

$$\delta \underline{C}_i = \varepsilon_{ii} \, \underline{C}_i \tag{7.1.1.3-2}$$

where

$\underline{C}_i$ = i^{th} column of $\left(C_B^L\right)^T$, which is the same as the i^{th} row of C_B^L, but in "official" column matrix vector format. It is unfortunate that we have to resort to such shenanigans for compatibility with the accepted practice of representing vectors as column rather than row matrices.

$\delta \underline{C}_i$ = Error in the i^{th} column of $\left(\hat{C}_B^L\right)^T$.

ε_{ii} = Element in row i, column i of E_{SYM}.

The normality error in $\hat{\underline{C}}_i$ is then corrected by subtracting $\delta \underline{C}_i$ as defined by (7.1.1.3-2) from $\hat{\underline{C}}_i$. Using the approximation that $\hat{\underline{C}}_i \approx \underline{C}_i$ obtains:

$$\hat{\underline{C}}_{i+} = \hat{\underline{C}}_{i-} - \delta \underline{C}_i = \hat{\underline{C}}_{i-} - \varepsilon_{ii} \, \underline{C}_i \approx \hat{\underline{C}}_{i-} - \varepsilon_{ii} \, \hat{\underline{C}}_{i-} = \left(1 - \varepsilon_{ii}\right) \hat{\underline{C}}_{i-} \tag{7.1.1.3-3}$$

where

-, + = Designation for the parameter value before (-) and after (+) the correction.

The orthogonality error between rows i and j of $\hat{C}_B^L$ can be caused by either row i, row j, or both being rotated from their nominal orthogonal condition. With no other information available we assume it is equally likely that the orthogonality error can be in either row i or j (i.e., $\hat{\underline{C}}_i$ and $\hat{\underline{C}}_j$), hence, half the error value is assigned to each. To correct for non-orthogonality, $\hat{\underline{C}}_i$ and $\hat{\underline{C}}_j$ must be rotated by one half the orthogonality error in the proper direction (away from each for the angle between $\hat{\underline{C}}_i$ and $\hat{\underline{C}}_j$ less than 90 degrees; i.e., for a positive dot product between $\hat{\underline{C}}_i$ and $\hat{\underline{C}}_j$). Since the rows of C_B^L are perpendicular to one another, the orthogonality correction for $\hat{\underline{C}}_i$ rotated away from $\hat{\underline{C}}_j$ is equivalent to a negative rotation of $\hat{\underline{C}}_i$ about the third row vector $\hat{\underline{C}}_k$ through one half the orthogonality error. Using the previous definition for the E_{SYM} error matrix, applying generalized Equation (3.2.2-20), and recognizing that the orthogonality error is very small, then yields for the $\hat{\underline{C}}_i$ orthogonalization correction operation:

$$\hat{\underline{C}}_{i+} = \left[I - \varepsilon_{ij} \left(\underline{C}_k \times\right)\right] \hat{\underline{C}}_{i-} \approx \hat{\underline{C}}_{i-} - \varepsilon_{ij} \, \hat{\underline{C}}_{k-} \times \hat{\underline{C}}_{i-} \tag{7.1.1.3-4}$$

where

ε_{ij} = Element in row i, column j of E_{SYM} (or in row j, column i since E_{SYM} is symmetric) which, from Equation (3.5.1-15), is half the dot product between $\underline{\hat{C}}_i$ and $\underline{\hat{C}}_j$.

For row j, the orthogonality correction is the same magnitude but in the opposite direction:

$$\underline{\hat{C}}_{j+} = \underline{\hat{C}}_{j-} + \varepsilon_{ij}\, \underline{\hat{C}}_{k-} \times \underline{\hat{C}}_{j-} \tag{7.1.1.3-5}$$

But because $\underline{\hat{C}}_i$, $\underline{\hat{C}}_j$ and $\underline{\hat{C}}_k$ are mutually perpendicular (approximately):

$$\underline{\hat{C}}_k \times \underline{\hat{C}}_i \approx \underline{\hat{C}}_j \qquad\qquad \underline{\hat{C}}_k \times \underline{\hat{C}}_j \approx -\underline{\hat{C}}_i \tag{7.1.1.3-6}$$

Substituting (7.1.1.3-6) in (7.1.1.3-4) and (7.1.1.3-5) then yields the orthogonality correction algorithm for $\hat{C}_B^L$ rows i and j:

$$\underline{\hat{C}}_{i+} = \underline{\hat{C}}_{i-} - \varepsilon_{ij}\, \underline{\hat{C}}_{j-} \qquad\qquad \underline{\hat{C}}_{j+} = \underline{\hat{C}}_{j-} - \varepsilon_{ij}\, \underline{\hat{C}}_{i-} \tag{7.1.1.3-7}$$

The combined normalization and orthogonalization correction process is the composite of Equations (7.1.1.3-3) and (7.1.1.3-7) for the three rows (i.e., normalization for each row and orthogonalization of the three sets of rows taken two at a time):

$$\begin{aligned}
\underline{\hat{C}}_{1+} &= \underline{\hat{C}}_{1-} - \varepsilon_{11}\, \underline{\hat{C}}_{1-} - \varepsilon_{12}\, \underline{\hat{C}}_{2-} - \varepsilon_{13}\, \underline{\hat{C}}_{3-} \\
\underline{\hat{C}}_{2+} &= \underline{\hat{C}}_{2-} - \varepsilon_{12}\, \underline{\hat{C}}_{1-} - \varepsilon_{22}\, \underline{\hat{C}}_{2-} - \varepsilon_{23}\, \underline{\hat{C}}_{3-} \\
\underline{\hat{C}}_{3+} &= \underline{\hat{C}}_{3-} - \varepsilon_{13}\, \underline{\hat{C}}_{1-} - \varepsilon_{23}\, \underline{\hat{C}}_{2-} - \varepsilon_{33}\, \underline{\hat{C}}_{3-}
\end{aligned} \tag{7.1.1.3-8}$$

or in matrix form using the definitions for the $\underline{\hat{C}}_i$'s and recognizing that $\varepsilon_{ij} = \varepsilon_{ji}$:

$$\left(\hat{C}_{B_+}^L\right)^T = \left(\hat{C}_{B_-}^L\right)^T - \left(\hat{C}_{B_-}^L\right)^T E_{SYM} = \left(\hat{C}_{B_-}^L\right)^T \left(I - E_{SYM}\right) \tag{7.1.1.3-9}$$

Because E_{SYM} is symmetric, it equals its transpose. Thus, the transpose of (7.1.1.3-9) is:

$$\hat{C}_{B_+}^L = \left(I - E_{SYM}\right) \hat{C}_{B_-}^L \tag{7.1.1.3-10}$$

Equation (7.1.1.3-1) for E_{SYM} measurement and (7.1.1.3-9) for E_{SYM} correction constitute the algorithm for correcting orthogonality/normality error in the rows of $\hat{C}_B^L$. As has already been mentioned, if the rows of $\hat{C}_B^L$ are orthogonalized and normalized, then the columns of $\hat{C}_B^L$ will also be orthogonalized and normalized.

An alternative to the previous approach is to structure the algorithm to measure and correct the normality/orthogonality error in the columns of $\hat{C}_B^L$. From (3.5.1-17), the orthogonality/normality error in the columns of $\hat{C}_B^L$ are:

$$E'_{SYM} = \frac{1}{2}\left[\left(\hat{C}_B^L\right)^T \hat{C}_B^L - I\right] \tag{7.1.1.3-11}$$

where

E'_{SYM} = Error in the <u>columns</u> of $\hat{C}_B^L$ characterized by symmetry about the $\hat{C}_B^L$ diagonal. E'_{SYM} is a symmetric matrix. The E'_{SYM} off-diagonal element in row i, column j equals half the perpendicularity error between columns i and j of $\hat{C}_B^L$ (negative for $\hat{C}_B^L$ columns i and j being more than 90 degrees apart). The E'_{SYM} i^{th} diagonal element equals the normality error (error in magnitude) of $\hat{C}_B^L$ column i.

From Equation (3.5.1-18), the relationship between E'_{SYM} and E_{SYM} is:

$$E_{SYM} = \hat{C}_B^L \, E'_{SYM} \left(\hat{C}_B^L\right)^{-1} \tag{7.1.1.3-12}$$

Substituting (7.1.1.3-12) into (7.1.1.3-10) obtains the $\hat{C}_B^L$ update algorithm based on E'_{SYM} measurements:

$$\hat{C}_{B_+}^L = \left(I - E_{SYM}\right)\hat{C}_{B_-}^L = \left[I - \hat{C}_{B_-}^L \, E'_{SYM}\left(\hat{C}_{B_-}^L\right)^{-1}\right]\hat{C}_{B_-}^L = \left(\hat{C}_{B_-}^L - \hat{C}_{B_-}^L \, E'_{SYM}\right)$$

or

$$\hat{C}_{B_+}^L = \hat{C}_{B_-}^L \left(I - E'_{SYM}\right) \tag{7.1.1.3-13}$$

Equations (7.1.1.3-11) and (7.1.1.3-13) (based on E'_{SYM}) are an alternative to Equations (7.1.1.3-1) and (7.1.1.3-9) (based on E_{SYM}) for correcting orthogonality/normality error in $\hat{C}_B^L$. Either set achieves the overall result of normalizing and orthogonalizing both the rows and columns of $\hat{C}_B^L$. To conserve computational throughput, the E_{SYM} form has been utilized in past practice as a means for eliminating the need for both computing E_{SYM} and correcting $\hat{C}_B^L$ in the same computation cycle. For this approach to be valid (i.e., stable) E_{SYM} evaluated with (7.1.1.3-1) in one computation cycle must be present and undistorted compared to the E_{SYM} that would have been calculated at the later time when $\hat{C}_B^L$ is corrected (using (7.1.1.3-9). This implies that E_{SYM} must have a low rate of change. The expectation for E_{SYM} to change over time can be evaluated by looking at its derivative. From (3.5.1-25) (with the L Frame substituted for the generic A Frame) we have:

$$\dot{E}_{SYM} = E_{SYM} \left(\underline{\hat{\omega}}_{IL}^{L} \times \right) - \left(\underline{\hat{\omega}}_{IL}^{L} \times \right) E_{SYM} + \frac{1}{2} \left[\delta \dot{C}_{B_{Comp}}^{L} \left(\hat{C}_{B}^{L} \right)^{T} + \hat{C}_{B}^{L} \left(\delta \dot{C}_{B_{Comp}}^{L} \right)^{T} \right] \qquad (7.1.1.3\text{-}14)$$

where

$\delta \dot{C}_{B_{Comp}}^{L}$ = Computational error rate in $\hat{C}_{B}^{L}$ produced from algorithm error, programming error, and computer finite word-length round-off (or truncation).

Equation (7.1.1.3-14) shows that past E_{SYM} values generate E_{SYM} changes proportional to the product of E_{SYM} with $\underline{\hat{\omega}}_{IL}^{L}$, the L Frame angular rate relative to inertial space (I). For a properly designed locally level frame (see Section 4.5 for options), $\underline{\hat{\omega}}_{IL}^{L}$ will be small in magnitude (typically no larger than a few earth rates). As such, E_{SYM} calculated at one time and applied later will have the same effect on $\hat{C}_{B}^{L}$ as if it was applied at the instant it was calculated. In contrast, the Equations (7.1.1.3-11) and (7.1.1.3-13) E'_{SYM} approach has Equation (3.5.1-26) for its change rate:

$$\dot{E}'_{SYM} = E'_{SYM} \left(\underline{\hat{\omega}}_{IB}^{B} \times \right) - \left(\underline{\hat{\omega}}_{IB}^{B} \times \right) E'_{SYM} + \frac{1}{2} \left[\left(\hat{C}_{B}^{L} \right)^{T} \delta \dot{C}_{B_{Comp}}^{L} + \left(\delta \dot{C}_{B_{Comp}}^{L} \right)^{T} \hat{C}_{B}^{L} \right] \qquad (7.1.1.3\text{-}15)$$

Equation (7.1.1.3-15) shows that past E'_{SYM} values generate E'_{SYM} changes proportional to the product of E'_{SYM} with $\underline{\hat{\omega}}_{IB}^{B}$, the strapdown sensor axis B Frame angular rate relative to inertial space (I). Since the magnitude of $\underline{\hat{\omega}}_{IB}^{B}$ can be large (under maneuvering conditions), past values of E'_{SYM} can be significantly distorted from the time it is measured until the time it is corrected. Hence, if the E'_{SYM} approach is to be applied (orthogonalization and normalization of the $\hat{C}_{B}^{L}$ columns), the time between measurement and correction must be short to preserve stability (e.g., in the same computation cycle).

It is also noted in passing that selection of the Equations (7.1.1.3-1) and (7.1.1.3-9) E_{SYM} form (orthogonalization/normalization of the $\hat{C}_{B}^{L}$ rows) facilitates the option of only computing two rows of $\hat{C}_{B}^{L}$ by integration, with the third row then calculated as the cross-product between the first two rows (i.e., as in Equations (4.1-9)). This is achieved by using Equations (7.1.1.3-3) and (7.1.1.3-7) to normalize and orthogonalize the first two rows, with the required E_{SYM} components calculated from (3.5.1-15):

$$\varepsilon_{ii} = \frac{1}{2} \left(\underline{\hat{C}}_{i} \cdot \underline{\hat{C}}_{i} - 1 \right) \qquad \varepsilon_{ij} = \frac{1}{2} \underline{\hat{C}}_{i} \cdot \underline{\hat{C}}_{j} \qquad (7.1.1.3\text{-}16)$$

The cross-product operation to obtain the third row will then automatically make it normal and orthogonal to rows one and two, thereby producing a complete $\hat{C}_B^L$ matrix with orthogonal and normal rows (and columns).

7.1.2 ATTITUDE QUATERNION (B TO L) UPDATE ALGORITHMS

The updating algorithm for the q_B^L attitude quaternion is developed following the identical procedure used for the C_B^L updating algorithm derivation in Section 7.1.1. We first adopt the identical nomenclature where:

$B_{I(m)}$ = Discrete attitude of the B Frame in non-rotating inertial space (I) at computer update time t_m.

m = Computer cycle index for B Frame motion updates to q_B^L.

$L_{I(n)}$ = Discrete attitude of the L Frame in non-rotating inertial space (I) at computer update time t_n.

n = Computer cycle index for L Frame motion updates to q_B^L.

The general updating algorithm for q_B^L is then constructed using the Equation (3.2.4.1-9) attitude quaternion chain rule:

$$
\begin{aligned}
q_{B_{I(m)}}^{L_{I(n-1)}} &= q_{B_{I(m-1)}}^{L_{I(n-1)}} \, q_{B_{I(m)}}^{B_{I(m-1)}} \\
q_{B_{I(m)}}^{L_{I(n)}} &= q_{L_{I(n-1)}}^{L_{I(n)}} \, q_{B_{I(m)}}^{L_{I(n-1)}}
\end{aligned}
\qquad (7.1.2\text{-}1)
$$

where

$q_{B_{I(m-1)}}^{L_{I(n-1)}}$ = q_B^L relating the B Frame at time t_{m-1} to the L Frame at time t_{n-1}.

$q_{B_{I(m)}}^{L_{I(n)}}$ = q_B^L relating the B Frame at time t_m to the L Frame at time t_n.

$q_{B_{I(m)}}^{B_{I(m-1)}}$ = Attitude quaternion that accounts for B Frame rotation relative to inertial space from its attitude at time t_{m-1} to its attitude at time t_m

$q_{L_{I(n-1)}}^{L_{I(n)}}$ = Attitude quaternion that accounts for L Frame rotation relative to inertial space from its attitude at time t_{n-1} to its attitude at time t_n.

The updates for q_B^L are performed by the $q_{B_{I(m)}}^{B_{I(m-1)}}$ and $q_{L_{I(n-1)}}^{L_{I(n)}}$ terms in (7.1.2-1), algorithm derivations for which follow.

7.1.2.1 BODY (B) FRAME UPDATE

The first expression in Equations (7.1.2-1) updates the q_B^L attitude quaternion using $q_{B_{I(m)}}^{B_{I(m-1)}}$ to account for angular rotation rate $\underline{\omega}_{IB}^B$ of the strapdown sensor (body) B Frame relative to non-rotating space:

$$q_{B_{I(m)}}^{L_{I(n-1)}} = q_{B_{I(m-1)}}^{L_{I(n-1)}} \, q_{B_{I(m)}}^{B_{I(m-1)}} \qquad (7.1.2.1\text{-}1)$$

with, formally:

$$q_{B_{I(m)}}^{B_{I(m-1)}} = q_I + \int_{t_{m-1}}^{t_m} \dot{q}_{B(t)}^{B_{I(m-1)}} \, dt \qquad (7.1.2.1\text{-}2)$$

where

q_I = Identity quaternion defined in Equation (3.3.4-4).

$q_{B(t)}^{B_{I(m-1)}}$ = Attitude quaternion relating the B Frame attitude at an arbitrary time t in the interval t_{m-1} to t_m, to its $B_{I(m)}$ attitude.

The $q_{B_{I(m)}}^{B_{I(m-1)}}$ attitude quaternion can also be expressed in terms of the rotation vector defining the Frame $B_{I(m)}$ attitude relative to Frame $B_{I(m-1)}$. Applying Equation (3.2.4.4-2) obtains:

$$q_{B_{I(m)}}^{B_{I(m-1)}} = \begin{bmatrix} \cos 0.5\ \phi_m \\ \dfrac{\sin 0.5\ \phi_m}{0.5\ \phi_m}\, 0.5\ \underline{\phi}_m \end{bmatrix}$$

$$\frac{\sin 0.5\ \phi_m}{0.5\ \phi_m} = 1 - \frac{(0.5\ \phi_m)^2}{3\,!} + \frac{(0.5\ \phi_m)^4}{5\,!} - \cdots \qquad (7.1.2.1\text{-}3)$$

$$\cos 0.5\ \phi_m = 1 - \frac{(0.5\ \phi_m)^2}{2\,!} + \frac{(0.5\ \phi_m)^4}{4\,!} - \cdots$$

where

$\underline{\phi}_m$ = Rotation vector defining the Frame $B_{I(m)}$ attitude relative to Frame $B_{I(m-1)}$ at time t_m.

ϕ_m = Magnitude of $\underline{\phi}_m$.

The $\underline{\phi}_m$ rotation vector in Equation (7.1.2.1-3) for attitude quaternion updating is identical to $\underline{\phi}_m$ used in Section 7.1.1.1 for C_B^L direction cosine matrix updating and is calculated using the identical algorithm provided by Equations (7.1.1.1-12), (7.1.1.1.1-17) and (7.1.1.1.1-18).

7.1.2.2 LOCAL LEVEL (L) FRAME ROTATION UPDATE

The second expression in Equations (7.1.2-1) updates the q_B^L attitude quaternion using $q_{L_{I(n-1)}}^{L_{I(n)}}$ to account for angular rotation rate $\underline{\omega}_{IL}^L$ of the local L Frame relative to non-rotating space:

$$q_{B_{I(m)}}^{L_{I(n)}} = q_{L_{I(n-1)}}^{L_{I(n)}} \, q_{B_{I(m)}}^{L_{I(n-1)}} \tag{7.1.2.2-1}$$

with, formally:

$$q_{L_{I(n-1)}}^{L_{I(n)}} = q_1 + \int_{t_{n-1}}^{t_n} \dot{q}_{L_{I(n-1)}}^{L(t)} \, dt \tag{7.1.2.2-2}$$

where

$q_{L_{I(n-1)}}^{L(t)}$ = Attitude quaternion relating the L Frame attitude at an arbitrary time t in the interval t_{n-1} to t_n, with its $L_{I(n-1)}$ attitude.

The $q_{L_{I(n-1)}}^{L_{I(n)}}$ attitude quaternion can also be expressed in terms of a rotation vector defining the Frame $L_{I(n)}$ attitude relative to Frame $L_{I(n-1)}$. Applying Equations (3.2.4.4-2) yields:

$$q_{L_{I(n-1)}}^{L_{I(n)}} = \begin{bmatrix} \cos 0.5\ \zeta_n \\ -\dfrac{\sin 0.5\ \zeta_n}{0.5\ \zeta_n}\ 0.5\ \underline{\zeta}_n \end{bmatrix}$$

$$\frac{\sin 0.5\ \zeta_n}{0.5\ \zeta_n} = 1 - \frac{(0.5\ \zeta_n)^2}{3\,!} + \frac{(0.5\ \zeta_n)^4}{5\,!} - \cdots$$

$$\cos 0.5\ \zeta_n = 1 - \frac{(0.5\ \zeta_n)^2}{2\,!} + \frac{(0.5\ \zeta_n)^4}{4\,!} - \cdots \qquad (7.1.2.2\text{-}3)$$

where

$\underline{\zeta}_n$ = Rotation vector defining the Frame $L_{I(n)}$ attitude at time t_n relative to Frame $L_{I(n-1)}$ attitude at time t_{n-1}.

ζ_n = Magnitude of $\underline{\zeta}_n$.

The negative sign on the $\dfrac{\sin 0.5\ \zeta_n}{0.5\ \zeta_n}\ 0.5\ \underline{\zeta}_n$ term accounts for the opposite phase sense of $q_{L_{I(n-1)}}^{L_{I(n)}}$ which describes the Frame $L_{I(n-1)}$ attitude relative to Frame $L_{I(n)}$ compared with the rotation vector $\underline{\zeta}_n$ phase sense which describes the Frame $L_{I(n)}$ attitude relative to Frame $L_{I(n-1)}$. The $\underline{\zeta}_n$ rotation vector in Equations (7.1.2.2-3) is identical to $\underline{\zeta}_n$ used for C_B^L direction cosine matrix updating and is calculated using the identical computational algorithm described in Section 7.1.1.2.1 and provided by Equations (7.1.1.2.1-3), (7.1.1.2.1-5) and (7.1.1.2.1-6).

An approximate form of Equations (7.1.2.2-3) that is comparable in accuracy to direction cosine updating Equation (7.1.1.2-5) is readily obtained by substitution and truncation:

$$q_{L_{I(n-1)}}^{L_{I(n)}} = \begin{bmatrix} 1 - \dfrac{1}{2}\,(\zeta_n/2)^2 \\ -\dfrac{1}{2}\,\underline{\zeta}_n \end{bmatrix} \qquad (7.1.2.2\text{-}4)$$

The comments in Section 7.1.1.2 regarding the advisability of using the simplified Equation (7.1.1.2-5) direction cosine local level frame updating algorithm also apply regarding use of Equation (7.1.2.2-4) for attitude quaternion updating rather than the complete Equations (7.1.2.2-3) form.

7.1.2.3 ATTITUDE QUATERNION (B TO L) NORMALIZATION CORRECTION

In order to preserve the fundamental attitude quaternion normality characteristic as represented in general by Equation (3.2.4.1-2), a normalization algorithm is frequently incorporated as an outer loop function in the q_B^L attitude quaternion updating process. The discussion at the beginning of Section 7.1.1.3 for direction cosine matrices regarding the need for a normalization/orthogonalization function is equally applicable for the attitude quaternion, the only exception being that "orthogonalization" has no meaning in the definition for the quaternion (as it does for the attitude direction cosine matrix), hence, the orthogonalization discussion in Section 7.1.1.3 does not apply.

To simplify notation, q will be utilized for q_B^L. We first write:

$$\hat{q} = q + \delta q \tag{7.1.2.3-1}$$

where

$\hat{\ }$ = Designation for parameter calculated in the INS computer, hence, containing error. The same parameter without the $\hat{\ }$ designation will be considered in this section to be the idealized error free value.

δq = Error in $\hat{q}$.

The normalization error in $\hat{q}$ can be determined from the requirement for the ideal q as given in Equations (3.2.4-21) and (3.2.4-22):

$$q\, q^* = 1 \tag{7.1.2.3-2}$$

Using (7.1.2.3-1), the product of $\hat{q}$ with its complex conjugate is:

$$\hat{q}\, \hat{q}^* = q\, q^* + q\, \delta q^* + \delta q\, q^* + \delta q\, \delta q^* \approx q\, q^* + q\, \delta q^* + \delta q\, q^* \tag{7.1.2.3-3}$$

We now define the δq in $\hat{q}$ due to abnormality as:

$$\delta q = \varepsilon_q\, q \tag{7.1.2.3-4}$$

where

ε_q = Normalization error in $\hat{q}$.

Substituting (7.1.2.3-4) in (7.1.2.3-3) with (7.1.2.3-2) obtains:

$$\hat{q}\,\hat{q}^* \;=\; q\,q^* + 2\,\varepsilon_q\,q\,q^* \;=\; \left(1 + 2\,\varepsilon_q\right) q\,q^* \;=\; 1 + 2\,\varepsilon_q \tag{7.1.2.3-5}$$

from which we see (using q_B^L for q by definition) that:

$$\varepsilon_q \;=\; \frac{1}{2}\left(\hat{q}_B^L\,\hat{q}_B^{L*} - 1\right) \tag{7.1.2.3-6}$$

The error in $\hat{q}$ is corrected by subtracting δq as defined by (7.1.2.3-4). Using the approximation that $\hat{q} \approx q$ obtains:

$$\hat{q} \;=\; \hat{q} - \;\delta q \;=\; \hat{q} - \;\varepsilon_q\,q \;\approx\; \hat{q} - \;\varepsilon_q\,\hat{q} \;=\; \left(1 - \varepsilon_q\right)\hat{q} \tag{7.1.2.3-7}$$

With q_B^L for q, (7.1.2.3-7) becomes the expression for quaternion normalization error correction:

$$\hat{q}_{B_+}^L \;=\; \left(1 - \varepsilon_q\right)\hat{q}_{B_-}^L \tag{7.1.2.3-8}$$

where

-, + = Designation for the parameter value before (-) and after (+) the correction.

7.1.2.4 QUATERNION TO DIRECTION COSINE MATRIX CONVERSION

If the quaternion is utilized for attitude determination, a conversion to the equivalent attitude direction cosine matrix form is typically incorporated for acceleration transformation and Euler angle extraction purposes. The appropriate conversion formula is provided by Equation (4.2-5):

$$C_B^L = \begin{bmatrix} \left(a^2 + b^2 - c^2 - d^2\right) & 2\,(b\,c - a\,d) & 2\,(b\,d + a\,c) \\ 2\,(b\,c + a\,d) & \left(a^2 - b^2 + c^2 - d^2\right) & 2\,(c\,d - a\,b) \\ 2\,(b\,d - a\,c) & 2\,(c\,d + a\,b) & \left(a^2 - b^2 - c^2 + d^2\right) \end{bmatrix} \tag{7.1.2.4-1}$$

where

a, b, c, d = Elements of q_B^L.

7.2 VELOCITY UPDATE ALGORITHMS

The navigation velocity algorithm calculates the velocity of the system relative to the earth fixed frame. The algorithm integrates specific force acceleration sensed by the body mounted accelerometers, Coriolis accelerations due to rotations of the local-level and earth frames, and gravity. The velocity algorithm implemented in the navigation software is formulated as a digital integration of Section 4.3 Equation (4.3-18) with (4.2-1) and (4.2-3) combined below:

$$\dot{\underline{v}}^N = C_L^N \, C_B^L \, \underline{a}_{SF}^B + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N \tag{7.2-1}$$

where

$\underline{v}^N$ = Velocity relative to the earth in N Frame axes.

$\underline{a}_{SF}^B$ = Specific force acceleration measured in the strapdown sensor body B Frame (by accelerometers).

$\underline{\omega}_{EN}^N$ = Angular rate of the local level navigation N Frame with respect to the earth E Frame (i.e., “transport rate”) in N Frame axes.

$\underline{\omega}_{IE}^N$ = Angular rate of the earth E Frame with respect to inertial space (I) in N Frame axes.

$\underline{g}_P^N$ = Plumb-bob gravity vector in the N Frame.

C_B^L = Direction cosine matrix between the B Frame and the local level L Frame.

C_L^N = Direction cosine matrix between the L and N Frame (see Equation (4.1.1-2)).

The digital velocity integration algorithm is formulated as:

$$\underline{v}_m^N = \underline{v}_{m-1}^N + C_L^N \, \Delta\underline{v}_{SF_m}^L + \Delta\underline{v}_{G/COR_m}^N \tag{7.2-2}$$

$$\Delta\underline{v}_{G/COR_m}^N = \int_{t_{m-1}}^{t_m} \left[\underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N\right] dt \tag{7.2-3}$$

$$\Delta\underline{v}_{SF_m}^L = \int_{t_{m-1}}^{t_m} C_B^L \, \underline{a}_{SF}^B \, dt \tag{7.2-4}$$

where

m = Digital integration algorithm update rate cycle index.

$\Delta\underline{v}_{SF_m}^L$ = L Frame coordinate portion of velocity change produced by specific force.

If vertical channel control is to be incorporated as in Equation (4.4.1.2.1-1) - (4.4.1.2.1-3), the following additional velocity update operation would be included representing the integral of the previous equations over an altitude update cycle:

$$\underline{v}_{n+}^{N} = \underline{v}_{n-}^{N} - e_{vc1_n} T_n \, \underline{u}_{ZN}^{N} \tag{7.2-5}$$

with

$$\partial h_n = h_n - h_{Prsr\,n}$$
$$e_{vc3_n} = e_{vc3_{n-1}} + C_1 \, \partial h_n \, T_n \qquad e_{vc1_n} = e_{vc3_n} + C_2 \, \partial h_n \qquad e_{vc2_n} = C_3 \, \partial h_n \tag{7.2-6}$$

where

-, + = Indicators for the value of $\underline{v}_n^N$ before (-) and after (+) the vertical stabilization addition.

n = Altitude update cycle index.

T_n = Time interval between altitude update cycles.

h_{Prsr_n} = Pressure altitude input signal.

e_{vc1_n} = Vertical velocity control signal.

e_{vc2_n} = Altitude control signal (See Equation (7.3.1-5) for how applied).

e_{vc3_n} = Integral controller.

C_1, C_2, C_3 = Vertical channel control gains (See Equations (4.4.1.2.1-11) for how typically determined).

Digital algorithms are formulated below for the (7.2-3) gravity/Coriolis velocity increment $\Delta\underline{v}_{G/COR_m}^{N}$ and the (7.2-4) integrated transformed specific force acceleration increment $\Delta\underline{v}_{SF_m}^{L}$. The algorithm developed for $\Delta\underline{v}_{SF_m}^{L}$ is modeled after the Section 7.1.1 two-speed attitude update approach; a moderate speed attitude update algorithm was used that is analytically exact under conditions when the angular rate vector is not rotating (i.e., constant in direction) during the attitude update time interval; a high speed algorithm was then used to measure rectifying rotating angular rate vector effects (i.e., "coning") for input to the moderate speed algorithm. For the $\Delta\underline{v}_{SF_m}^{L}$ two-speed update approach, the moderate speed algorithm is designed to be analytically exact under constant B Frame angular-rate/specific-force-acceleration vector direction and magnitude ratio during the velocity update period; the high speed algorithm measures rectifying dynamic angular-rate/linear-acceleration effects ("sculling") for input to the moderate speed algorithm.

7.2.1 GRAVITY/CORIOLIS VELOCITY INCREMENT ALGORITHM

From Equations (5.4.1-11) with (5.4.1-9), (5.4-4), (5.4-1), and (5.4-2), the $\underline{g}_P^N$ term in Equation (7.2-3) is a function of position location with very small horizontal components. Since the position varies smoothly over a digital algorithm m cycle with limited magnitude change (particularly in altitude), $\underline{g}_P^N$ in (7.2-3) can be approximated by its average value across the m cycle. Because the (7.2-3) Coriolis term (angular rate products with velocity) is small (due to the small size of the angular rates) and because velocity varies smoothly over an m cycle, the Coriolis contributors can also be approximated by their average value over the m cycle. This rationale forms the basis for the following trapezoidal integration algorithm typically utilized for $\Delta\underline{v}_{G/COR_m}^N$ in (7.2-3) using (4.1.1-6) for $\underline{\omega}_{EN}^N$:

$$\begin{aligned}\Delta\underline{v}_{G/COR_m}^N \approx \Big\{\underline{g}_{P_{m-1/2}}^N - \Big[2\,\underline{\omega}_{IE_{m-1/2}}^N + \rho_{ZN_{m-1/2}}\,\underline{u}_{ZN}^N \\ + F_{C_{m-1/2}}^N\left(\underline{u}_{ZN}^N \times \underline{v}_{m-1/2}^N\right)\Big] \times \underline{v}_{m-1/2}^N\Big\}\,T_m\end{aligned} \qquad (7.2.1\text{-}1)$$

where

$m\text{-}1/2$ = Designation for parameter value midway between t_{m-1} and t_m.

T_m = Velocity integration algorithm update period $t_m - t_{m-1}$.

The $\underline{\omega}_{IE}^N$ term in (7.2.1-1) is evaluated with Equation (4.1.1-3), ρ_{ZN} is computed based on N Frame selection in Section 4.5, and $\underline{g}_P^N$ is calculated from Section 5.4.1, Equation (5.4.1-11) and its inputs.

Because $\Delta\underline{v}_{G/COR_m}^N$ is used in Equation (7.2-2) to update $\underline{v}^N$ from its m-1 to m cycle value, $\underline{v}_{m-1/2}^N$ is not explicitly available for Equation (7.2.1-1) and must be approximated based on extrapolation from past values. An example is the linear extrapolation algorithm:

$$\underline{v}_{m-1/2}^N \approx \underline{v}_{m-1}^N + \frac{1}{2}\left[\underline{v}_{m-1}^N - \underline{v}_{m-2}^N\right] = \frac{3}{2}\,\underline{v}_{m-1}^N - \frac{1}{2}\,\underline{v}_{m-2}^N \qquad (7.2.1\text{-}2)$$

The $\underline{g}_P^N$, $\underline{\omega}_{IE}^N$, ρ_{ZN}, F_C parameters in (7.2.1-1) are functions of position which (from Section 7.3) is updated following the velocity update, possibly at a slower n cycle computation rate.

Therefore, the m-1/2 designation evaluation for these parameters is not explicitly available and must also be approximated based on extrapolation from past values. For example, for linear extrapolation:

$$(\ )_{m-1/2} \approx (\ )_{n-1} + \frac{(r - 1/2)}{j}\left[(\ )_{n-1} - (\ )_{n-2}\right] \tag{7.2.1-3}$$

where

n = Computer cycle index for position updates.

j = Number of m cycles in each n cycle.

r = Number of m cycles at time t_m since the last n cycle (i.e., since t_{n-1}).

7.2.2 INTEGRATED TRANSFORMED SPECIFIC FORCE ACCELERATION INCREMENT ALGORITHM

A digital algorithm for $\Delta \underline{v}_{SF_m}^{L}$ integrated transformed acceleration increment Equation (7.2-4) must account for the rotation of the local level L Frame and the strapdown sensor "body" B Frame during the t_{m-1} to t_m computation period. Adopting the same notation used in Section 7.1.1 to describe discrete attitudes of the L and B Frames relative to inertial space (I) at computer update time instants, Equation (7.2-4) can be expanded using the Equation (3.2.1-5) chain rule and integration by parts as follows:

$$\begin{aligned}
\Delta \underline{v}_{SF_m}^{L} = \Delta \underline{v}_{SF_m}^{L_{I(m)}} &= \int_{t_{m-1}}^{t_m} C_{L_{I(m-1)}}^{L(t)}\, C_{L_{I(n-1)}}^{L_{I(m-1)}}\, C_{B_{I(m-1)}}^{L_{I(n-1)}}\, C_{B(t)}^{B_{I(m-1)}}\, \underline{a}_{SF}^{B}\, dt \\
&= C_{L_{I(m-1)}}^{L_{I(m)}}\, C_{L_{I(n-1)}}^{L_{I(m-1)}}\, C_{B_{I(m-1)}}^{L_{I(n-1)}} \int_{t_{m-1}}^{t_m} C_{B(t)}^{B_{I(m-1)}}\, \underline{a}_{SF}^{B}\, dt \\
&\quad - \int_{t_{m-1}}^{t_m} \dot{C}_{L_{I(m-1)}}^{L(t)}\, C_{L_{I(n-1)}}^{L_{I(m-1)}}\, C_{B_{I(m-1)}}^{L_{I(n-1)}} \int_{t_{m-1}}^{t} C_{B(\tau)}^{B_{I(m-1)}}\, \underline{a}_{SF}^{B}\, d\tau\, dt
\end{aligned} \tag{7.2.2-1}$$

Assuming $\dot{C}_{L_{I(m-1)}}^{L(t)}$ is approximately constant and $\int_{t_{m-1}}^{t} C_{B(\tau)}^{B_{I(m-1)}}\, \underline{a}_{SF}^{B}\, d\tau$ changes approximately linearly over the m cycle we then get:

$$
\begin{aligned}
\Delta\underline{v}_{SF_m}^{L} &\approx C_{L_{I(m-1)}}^{L_{I(m)}} \, C_{L_{I(n-1)}}^{L_{I(m-1)}} \, C_{B_{I(m-1)}}^{L_{I(n-1)}} \int_{t_{m-1}}^{t_m} C_{B(t)}^{B_{I(m-1)}} \, \underline{a}_{SF}^{B} \, dt \\
&\quad - \int_{t_{m-1}}^{t_m} \frac{1}{T_m} \left(C_{L_{I(m-1)}}^{L_{I(m)}} - I \right) C_{L_{I(n-1)}}^{L_{I(m-1)}} \, C_{B_{I(m-1)}}^{L_{I(n-1)}} \left(\left(\int_{t_{m-1}}^{t_m} C_{B(\tau)}^{B_{I(m-1)}} \, \underline{a}_{SF}^{B} \, d\tau \right) \frac{(t - t_{m-1})}{T_m} \right) dt \\
&= C_{L_{I(m-1)}}^{L_{I(m)}} \, C_{L_{I(n-1)}}^{L_{I(m-1)}} \, C_{B_{I(m-1)}}^{L_{I(n-1)}} \int_{t_{m-1}}^{t_m} C_{B(t)}^{B_{I(m-1)}} \, \underline{a}_{SF}^{B} \, dt \\
&\quad - \frac{1}{2} \left(C_{L_{I(m-1)}}^{L_{I(m)}} - I \right) C_{L_{I(n-1)}}^{L_{I(m-1)}} \, C_{B_{I(m-1)}}^{L_{I(n-1)}} \int_{t_{m-1}}^{t_m} C_{B(t)}^{B_{I(m-1)}} \, \underline{a}_{SF}^{B} \, dt \qquad (7.2.2\text{-}1a) \\
&= \frac{1}{2} \left(C_{L_{I(m-1)}}^{L_{I(m)}} + I \right) C_{L_{I(n-1)}}^{L_{I(m-1)}} \, C_{B_{I(m-1)}}^{L_{I(n-1)}} \int_{t_{m-1}}^{t_m} C_{B(t)}^{B_{I(m-1)}} \, \underline{a}_{SF}^{B} \, dt \\
&= \frac{1}{2} \left(C_{L_{I(n-1)}}^{L_{I(m)}} + C_{L_{I(n-1)}}^{L_{I(m-1)}} \right) C_{B_{I(m-1)}}^{L_{I(n-1)}} \int_{t_{m-1}}^{t_m} C_{B(t)}^{B_{I(m-1)}} \, \underline{a}_{SF}^{B} \, dt
\end{aligned}
$$

in which it is recognized that the L and L_I Frames at time t_m are identical by definition of the L_I Frame. Upon further expansion:

$$
\Delta\underline{v}_{SF_m}^{L_{I(n-1)}} = C_{B_{I(m-1)}}^{L_{I(n-1)}} \, \Delta\underline{v}_{SF_m}^{B_{I(m-1)}} \qquad (7.2.2\text{-}2)
$$

$$
\Delta\underline{v}_{SF_m}^{B_{I(m-1)}} = \int_{t_{m-1}}^{t_m} C_{B(t)}^{B_{I(m-1)}} \, \underline{a}_{SF}^{B} \, dt \qquad (7.2.2\text{-}3)
$$

$$
\begin{aligned}
\Delta\underline{v}_{SF_m}^{L} &\approx \frac{1}{2} \left(C_{L_{I(n-1)}}^{L_{I(m)}} + C_{L_{I(n-1)}}^{L_{I(m-1)}} \right) \Delta\underline{v}_{SF_m}^{L_{I(n-1)}} \\
&= \Delta\underline{v}_{SF_m}^{L_{I(n-1)}} + \frac{1}{2} \left[\left(C_{L_{I(n-1)}}^{L_{I(m)}} - I \right) + \left(C_{L_{I(n-1)}}^{L_{I(m-1)}} - I \right) \right] \Delta\underline{v}_{SF_m}^{L_{I(n-1)}}
\end{aligned} \qquad (7.2.2\text{-}4)
$$

As in Section 7.1.1, Equations (7.2.2-2) - (7.2.2-4) allow for the general case in which the C_B^L matrix is updated for L Frame rotation at a cycle rate (index n) that may differ (be slower) than the C_B^L update rate for B Frame rotation (index m). For example, in the interests of minimizing computer throughput requirements, the software architecture might have the n cycle L Frame update rate 5 times slower than the m cycle B Frame update rate. Equations (7.2.2-2)

- (7.2.2-4) are also valid, however, if we choose to update C_B^L at equal rates for B and L Frame motion (i.e., $n = m$). Note, that for $n \neq m$, Equation (7.2.2-4) still requires an L Frame orientation evaluation at the B Frame m cycle update time (for $L_{I(m)}$ in the $C_{L_{I(n-1)}}^{L_{I(m)}}$ matrix). Note also, that the form of the (7.2.2-2) algorithm is based on the use of C_B^L at the previous B Frame m cycle (i.e., $B_{I(m-1)}$ in the $C_{B_{I(m-1)}}^{L_{I(n-1)}}$ matrix). This implies that C_B^L will be updated for B Frame rotation following the Equation (7.2.2-2) acceleration transformation operation.

It remains to define algorithms for calculating $C_{L_{I(n-1)}}^{L_{I(m)}}$ in (7.2.2-4) to correct for local level frame rotation during acceleration transformation, and for the $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ body frame specific force acceleration increment term in (7.2.2-2) and (7.2.2-3).

7.2.2.1 CORRECTION FOR L FRAME ROTATION DURING ACCELERATION TRANSFORMATION

The $C_{L_{I(n-1)}}^{L_{I(m)}}$ matrix in (7.2.2-4) can be defined analytically as in Equations (7.1.1.2-3) and (7.1.1.2.1-3) - (7.1.1.2.1-6), but with the $\underline{\zeta}_n$ term calculated as a summation from t_{n-1} to t_m rather than from t_{n-1} to t_n. Because of the very small angular rotation rate of the L Frame relative to inertial space, $C_{L_{I(n-1)}}^{L_{I(m)}}$ in (7.2.2-4) is very close to the identity matrix I. For many applications, $\frac{1}{2}\left[\left(C_{L_{I(n-1)}}^{L_{I(m)}} - I\right) + \left(C_{L_{I(n-1)}}^{L_{I(m-1)}} - I\right)\right] \Delta \underline{v}_{SF_m}^{L_{I(n-1)}}$ in (7.2.2-4) can, therefore, be totally ignored as negligible compared to other acceleration error sources. For high accuracy applications when $\frac{1}{2}\left[\left(C_{L_{I(n-1)}}^{L_{I(m)}} - I\right) + \left(C_{L_{I(n-1)}}^{L_{I(m-1)}} - I\right)\right] \Delta \underline{v}_{SF_m}^{L_{I(n-1)}}$ is to be included, a first order form of (7.1.1.2-3) - (7.1.1.2-4) usually suffices such that:

$$C_{L_{I(n-1)}}^{L_{I(m)}} \approx I - \left(\underline{\zeta}_{n-1,m} \times\right) \tag{7.2.2.1-1}$$

$$\underline{\zeta}_{n-1,m} = \int_{t_{n-1}}^{t_m} \underline{\omega}_{IL}^L \, dt \tag{7.2.2.1-2}$$

We then approximate $\underline{\omega}_{IL}^L$ in (7.2.2.1-2) using (4.1.1-1), (4.1.1-6), and the assumption of slowly changing contributors as in Section 7.2.1:

$$\underline{\omega}_{IL}^L = C_N^L \left(\underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N\right) \approx C_N^L \left[\underline{\omega}_{IE_{\frac{n-1,m}{2}}}^N + \rho_{ZN_{\frac{n-1,m}{2}}} \underline{u}_{ZN}^N + F_{C_{\frac{n-1,m}{2}}}^N \left(\underline{u}_{ZN}^N \times \underline{v}^N\right)\right] \tag{7.2.2.1-3}$$

where

$\frac{n-1,m}{2}$ = Subscript indicating value for parameter midway between the times t_{n-1} and t_m.

Substituting (7.2.2.1-3) into (7.2.2.1-2) yields for $\underline{\zeta}_{n-1,m}$:

$$\underline{\zeta}_{n-1,m} \approx C_N^L \left[\underline{\omega}^N_{IE_{\frac{n-1,m}{2}}} \, r \, T_m + \rho_{ZN_{\frac{n-1,m}{2}}} \, \underline{u}^N_{ZN} \, r \, T_m + F^N_{C_{\frac{n-1,m}{2}}} \left(\underline{u}^N_{ZN} \times \Delta \underline{R}^N_{n-1,m} \right) \right] \tag{7.2.2.1-4}$$

$$\Delta \underline{R}^N_{n-1,m} \equiv \int_{t_{n-1}}^{t_m} \underline{v}^N \, dt \tag{7.2.2.1-5}$$

where

r = Number of m cycles at time t_m since the last n cycle (i.e., since t_{n-1}).

T_m = Time interval between m cycles.

The $\underline{\omega}^N_{IE}$ term in (7.2.2.1-4) is evaluated with Equation (4.1.1-3) and ρ_{ZN} is calculated based on N Frame selection in Section 4.5.

As in Section 7.2.1, $\left(\, \right)_{\frac{n-1,m}{2}}$ in (7.2.2.1-4) must be approximated based on past value extrapolation; e.g.:

$$\left(\, \right)_{\frac{n-1,m}{2}} \approx \left[\left(\, \right)_{n-1} + \frac{1}{2} \, \frac{r}{j} \left(\left(\, \right)_{n-1} - \left(\, \right)_{n-2} \right) \right] \tag{7.2.2.1-6}$$

where

j = Number of m cycles in each n cycle.

Because (7.2.2.1-4) is used to update $\underline{v}^N$ in Equations (7.2-2) with (7.2.2-4) and (7.2.2.1-1), current values of $\underline{v}^N$ are not available for evaluating $\Delta \underline{R}^N_{n-1,m}$ in (7.2.2.1-5). Hence, past value extrapolation must be employed such as in Section 7.2.1:

$$\Delta \underline{R}^N_{n-1,m} \approx \frac{T_m}{2} \left(3 \, \underline{v}^N_{m-1} - \underline{v}^N_{m-2} \right) \qquad \text{For } r = 1$$
$$\Delta \underline{R}^N_{n-1,m} \approx \frac{T_m}{2} \left[3 \, \underline{v}^N_{m-1} - \underline{v}^N_{m-2} + \sum_{i = m+1-r}^{i = m-1} \left(\underline{v}^N_i + \underline{v}^N_{i-1} \right) \right] \qquad \text{For } r > 1 \tag{7.2.2.1-7}$$

7.2.2.2 BODY FRAME INTEGRATED SPECIFIC FORCE ACCELERATION INCREMENT

The $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ integral term in (7.2.2-2) and (7.2.2-3) is calculated using a high speed digital algorithm similar to the type employed in Equations (7.1.1.1-12) based on (7.1.1.1-13) for attitude updating. Derivation of the algorithm is initially based on first order approximations for $C_{B(t)}^{B_{I(m-1)}}$. The first order solution is then divided into two parts for application of the "two-speed" algorithm approach; a portion that measures dynamic B Frame angular-rate/specific-force-acceleration through high speed integration within the m cycle, and a portion calculated at the m cycle rate that measures the effect of slowly changing B Frame angular-rate/specific-force. Finally, the first order m cycle portion is expanded to be exact for constant B Frame angular-rate/specific-force vector direction and constant ratio of the angular-rate/specific-force-acceleration vector magnitudes (of which a special case is constant B Frame angular rate/specific force direction and magnitude).

Following the similar development incorporated in Section 7.1.1.1, the $C_{B(t)}^{B_{I(m-1)}}$ term in the Equation (7.2.2-3) $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ integrand is expressed as:

$$C_{B(t)}^{B_{I(m-1)}} = I + \frac{\sin \phi(t)}{\phi(t)} \left(\underline{\phi}(t) \times \right) + \frac{1 - \cos \phi(t)}{\phi(t)^2} \left(\underline{\phi}(t) \times \right)^2 \qquad (7.2.2.2\text{-}1)$$

where

$\underline{\phi}(t)$ = Rotation vector defining the general attitude of Frame B relative to Frame $B_{I(m-1)}$ for time t greater than t_{m-1}.

$\phi(t)$ = Magnitude of $\underline{\phi}(t)$.

Equations (7.1.1.1-9) and (7.1.1.1-10) show that for $\underline{\alpha}(t)$ small compared to one (i.e., a small coning effect), $\underline{\phi}(t)$ in (7.2.2.2-1) can be approximated by:

$$\underline{\phi}(t) \approx \underline{\alpha}(t) \qquad (7.2.2.2\text{-}2)$$

with

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}^{B} \, d\tau \qquad (7.2.2.2\text{-}3)$$

A first order approximation for Equation (7.2.2.2-1) neglects $\left(\underline{\phi}(t)\times\right)^2$ and approximates $\frac{\sin\phi(t)}{\phi(t)}$ by unity (assuming that the m cycle rate is selected fast enough to maintain $\underline{\phi}(t)$ at a reasonably small value; e.g., less than 0.05 radians). With (7.2.2.2-2), Equation (7.2.2.2-1) reduces to:

$$C^{B_{I(m-1)}}_{B(t)} \approx I + \left(\underline{\alpha}(t)\times\right) \tag{7.2.2.2-4}$$

Substituting (7.2.2.2-4) into (7.2.2-3) then yields to first order:

$$\Delta\underline{v}^{B_{I(m-1)}}_{SF_m} = \int_{t_{m-1}}^{t_m} \left[I + \left(\underline{\alpha}(t)\times\right)\right] \underline{a}^B_{SF}\,dt = \int_{t_{m-1}}^{t_m} \underline{a}^B_{SF}\,dt + \int_{t_{m-1}}^{t_m} \left(\underline{\alpha}(t)\times\right) \underline{a}^B_{SF}\,dt \tag{7.2.2.2-5}$$

or

$$\Delta\underline{v}^{B_{I(m-1)}}_{SF_m} = \underline{\upsilon}_m + \int_{t_{m-1}}^{t_m} \left(\underline{\alpha}(t)\times \underline{a}^B_{SF}\right) dt$$

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}^B_{IB}\,d\tau \qquad \underline{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{a}^B_{SF}\,d\tau \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m) \tag{7.2.2.2-6}$$

where

$\underline{\upsilon}(t)$ = Integral of $\underline{a}^B_{SF}$ from time t_{m-1} to time t.

Equations (7.2.2.2-6) define a method for calculating $\Delta\underline{v}^{B_{I(m-1)}}_{SF_m}$ for Equation (7.2.2-2). It is instructive to analyze these equations under conditions when the angular rate and specific force vectors are not rotating in the B Frame, and the ratio of magnitudes of the angular rate and specific force vectors is constant. This condition covers the case when the B Frame angular rate and specific force vectors are both constant, but also covers the broader case when the angular-rate/specific-force vectors have constant B Frame direction with identical time function magnitudes (e.g., in-phase sinusoidal oscillations of the angular rate and specific force magnitudes). Under the conditions described above, we can write for the angular rate vector $\underline{\omega}^B_{IB}$ and its integral:

$$\underline{\omega}^B_{IB} = \omega\,\underline{u}_\omega \qquad \underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}^B_{IB}\,d\tau = \underline{u}_\omega \int_{t_{m-1}}^{t} \omega\,d\tau = \underline{u}_\omega\,\alpha(t) \tag{7.2.2.2-7}$$

where

ω = Magnitude of $\underline{\omega}_{IB}^{B}$.

$\underline{u}_{\omega}$ = Unit vector along $\underline{\omega}_{IB}^{B}$ in which $\underline{u}_{\omega}$ is constant in the B Frame.

$\alpha(t)$ = Magnitude of $\underline{\alpha}(t)$.

We also note from (7.2.2.2-7) that:

$$\underline{u}_{\omega}\, d\alpha \;=\; \underline{\omega}_{IB}^{B}\, dt \;=\; \underline{u}_{\omega}\, \omega\, dt \tag{7.2.2.2-8}$$

or

$$\omega\, dt \;=\; d\alpha \tag{7.2.2.2-9}$$

For the $\underline{a}_{SF}^{B}$ specific force vector, we can write for constant B Frame acceleration direction with magnitude proportional to the angular rate magnitude:

$$\underline{a}_{SF}^{B} \;=\; \underline{u}_{a}\, a_{SF} \qquad\qquad a_{SF} \;=\; K\,\omega \tag{7.2.2.2-10}$$

where

a_{SF} = Magnitude of $\underline{a}_{SF}^{B}$.

$\underline{u}_{a}$ = Unit vector along $\underline{a}_{SF}^{B}$ in which $\underline{u}_{a}$ is constant in the B Frame.

K = Constant equal to the ratio of a_{SF} to ω.

We make note for later use that from (7.2.2.2-6), (7.2.2.2-7) and (7.2.2.2-10):

$$\underline{\upsilon}(t) \;=\; \int_{t_{m-1}}^{t} \underline{a}_{SF}^{B}\, d\tau \;=\; \underline{u}_{a}\, K \int_{t_{m-1}}^{t} \omega\, d\tau \;=\; \underline{u}_{a}\, K\, \alpha(t) \tag{7.2.2.2-11}$$

Substituting (7.2.2.2-7), (7.2.2.2-9) and (7.2.2.2-10) into the (7.2.2.2-6) $\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$ integral term, finds using (7.1.1.1-13):

For Non-Rotating B Frame Angular-Rate/Specific-Force Vectors With Constant Angular-Rate/Specific-Force Magnitude Ratio:

$$
\begin{aligned}
\int_{t_{m-1}}^{t_m} \left(\underline{\alpha}(t) \times \underline{a}_{SF}^{B} \right) dt &= \int_{t_{m-1}}^{t_m} \left[\left(\underline{u}_{\omega}\, \alpha(t) \right) \times \left(\underline{u}_a\, K\, \omega \right) \right] dt \\
&= \left[\underline{u}_{\omega} \times \left(\underline{u}_a\, K \right) \right] \int_{t_{m-1}}^{t_m} \alpha(t)\, \omega\, dt = \left[\underline{u}_{\omega} \times \left(\underline{u}_a\, K \right) \right] \int_{\alpha = 0}^{\alpha_m} \alpha\, d\alpha \qquad (7.2.2.2\text{-}12) \\
&= \frac{1}{2} \left[\underline{u}_{\omega} \times \left(\underline{u}_a\, K \right) \right] \alpha^2 \Big]_{\alpha = 0}^{\alpha_m} = \frac{1}{2} \left[\underline{u}_{\omega} \times \left(\underline{u}_a\, K \right) \right] \alpha_m^2
\end{aligned}
$$

But from (7.2.2.2-6) with (7.2.2.2-7) and (7.2.2.2-10) we can also write:

$$
\begin{aligned}
\underline{\alpha}_m &= \int_{t_{m-1}}^{t_m} \underline{\omega}_{IB}^{B}\, dt = \underline{u}_{\omega} \int_{t_{m-1}}^{t_m} \omega\, dt = \underline{u}_{\omega}\, \alpha_m \\
\underline{\upsilon}_m &= \int_{t_{m-1}}^{t_m} \underline{a}_{SF}^{B}\, dt = \underline{u}_a \int_{t_{m-1}}^{t_m} a_{SF}\, dt = \underline{u}_a\, K \int_{t_{m-1}}^{t_m} \omega\, dt = \underline{u}_a\, K\, \alpha_m
\end{aligned}
\qquad (7.2.2.2\text{-}13)
$$

from which:

$$
\underline{u}_{\omega} = \underline{\alpha}_m / \alpha_m \qquad \underline{u}_a\, K = \underline{\upsilon}_m / \alpha_m \qquad (7.2.2.2\text{-}14)
$$

Substituting (7.2.2.2-12) with (7.2.2.2-14) into (7.2.2.2-6) then obtains:

For Non-Rotating B Frame Angular-Rate/Specific-Force Vectors With Constant Angular-Rate/Specific-Force Magnitude Ratio:

$$
\begin{aligned}
\Delta \underline{v}_{SF_m}^{B_{I(m-1)}} &= \underline{\upsilon}_m + \frac{1}{2}\, \underline{\alpha}_m \times \underline{\upsilon}_m \\
\underline{\alpha}(t) &= \int_{t_{m-1}}^{t} \underline{\omega}_{IB}^{B}\, d\tau \qquad \underline{\alpha}_m = \underline{\alpha}(t_m) \\
\underline{\upsilon}(t) &= \int_{t_{m-1}}^{t} \underline{a}_{SF}^{B}\, d\tau \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)
\end{aligned}
\qquad (7.2.2.2\text{-}15)
$$

If we now compare Equations (7.2.2.2-6) for the general case with Equations (7.2.2.2-15) for the particular case noted, we see that the difference is the replacement of the integral term with $\frac{1}{2}\,\underline{\alpha}_m \times \underline{\upsilon}_m$. For situations when non-rotating B Frame angular-rate/specific-force vectors with constant angular-rate/specific-force magnitude ratio is a reasonable approximation over the t_{m-1} to t_m time interval, Equations (7.2.2.2-15) are preferred because the integral term (and its attendant high speed algorithm) is replaced by $\frac{1}{2}\,\underline{\alpha}_m \times \underline{\upsilon}_m$ which is evaluated once at the m cycle time.

A fundamental limitation in Equations (7.2.2.2-6) or (7.2.2.2-15) is the first order approximation that underlies their development (i.e., Equation (7.2.2.2-4) for $C_{B(t)}^{BI(m-1)}$ that was used in the Equation (7.2.2-3) $\Delta\underline{v}_{SF_m}^{BI(m-1)}$ expression). It would be desirable if the Equation (7.2.2.2-4) approximation could be applied only to the high frequency content of $C_{B(t)}^{BI(m-1)}$, with the low frequency content retaining the full Equation (7.2.2.2-1) form. Such an algorithm can be synthesized by first noting that:

$$\frac{d}{dt}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) = \underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) + \dot{\underline{\alpha}}(t) \times \underline{\upsilon}(t) = \underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) - \underline{\upsilon}(t) \times \dot{\underline{\alpha}}(t) \qquad (7.2.2.2\text{-}16)$$

with $\underline{\alpha}(t)$ and $\underline{\upsilon}(t)$ as defined in Equations (7.2.2.2-6). Upon rearrangement, (7.2.2.2-16) becomes:

$$\underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) = \frac{d}{dt}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) + \underline{\upsilon}(t) \times \dot{\underline{\alpha}}(t) \qquad (7.2.2.2\text{-}17)$$

Trivially,

$$\underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) = \frac{1}{2}\,\underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) + \frac{1}{2}\,\underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) \qquad (7.2.2.2\text{-}18)$$

We now substitute (7.2.2.2-17) for one of the terms on the right in (7.2.2.2-18) to obtain:

$$\underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) = \frac{1}{2}\frac{d}{dt}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) + \frac{1}{2}\left(\underline{\alpha}(t) \times \dot{\underline{\upsilon}}(t) + \underline{\upsilon}(t) \times \dot{\underline{\alpha}}(t)\right) \qquad (7.2.2.2\text{-}19)$$

From Equations (7.2.2.2-6) we know that:

$$\dot{\underline{\alpha}}(t) = \underline{\omega}_{IB}^{B} \qquad \dot{\underline{\upsilon}}(t) = \underline{a}_{SF}^{B} \qquad (7.2.2.2\text{-}20)$$

with which Equation (7.2.2.2-19) assumes the form:

$$\underline{\alpha}(t) \times \underline{a}_{SF}^{B} = \frac{1}{2}\frac{d}{dt}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) + \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(t) \times \underline{\omega}_{IB}^{B}\right) \qquad (7.2.2.2\text{-}21)$$

Equation (7.2.2.2-21) is an alternate expression for the integrand in the Equations (7.2.2.2-6) $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ expression. Substituting (7.2.2.2-21) for the integrand yields the following equivalent form:

$$\Delta \underline{v}_{SF_m}^{B_{I(m-1)}} = \underline{\upsilon}_m + \frac{1}{2}\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) + \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(t) \times \underline{\omega}_{IB}^{B}\right) dt \qquad (7.2.2.2\text{-}22)$$

If we compare Equation (7.2.2.2-22) with Equations (7.2.2.2-15) for $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ under non-rotating B Frame angular-rate/specific-force vectors with constant angular-rate/specific-force magnitude ratio (of which constant angular-rate/specific-force vectors is a special case), we see that they are equivalent except for the integral term in (7.2.2.2-22). It is easily verified by substituting Equations (7.2.2.2-7), (7.2.2.2-10) and (7.2.2.2-11) into (7.2.2.2-22), that the integral term in (7.2.2.2-22) vanishes for non-rotating B Frame angular-rate/specific-force vectors with constant angular-rate/specific-force magnitude ratio. We conclude that the integral term in (7.2.2.2-22) represents a contribution from “high frequency” components in the $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ Equation (7.2.2-3) integrand, while the remaining terms (i.e., $\underline{\upsilon}_m + \frac{1}{2}\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right)$) represent a combination of both “low frequency" and “high frequency” effects.

The integral term in (7.2.2.2-22), denoted as “sculling”, measures the “constant” contribution to $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ created from combined dynamic angular-rate/specific-force rectification. The rectification is a maximum under classical sculling motion defined as sinusoidal angular-rate/specific-force in which the $\underline{\alpha}(t)$ angular excursion about one B Frame axis is at the same frequency and in phase with the $\underline{a}_{SF}^{B}$ specific force along another B Frame axis (with a constant acceleration component then produced along the average third axis direction). This is the same principle used by mariners to propel a boat in the forward direction using a single oar operated with an undulating motion (also denoted as “sculling", the original use of the term). Note, that the $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ integral term in Equations (7.2.2.2-6) has also been defined as “sculling” even though it contains large contributions under conditions of non-rotating B Frame angular-rate/specific-force vectors with constant angular-rate/specific-force magnitude ratio (i.e., “non-sculling” conditions). The $\frac{1}{2}\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right)$ term in (7.2.2.2-22) is denoted herein as “velocity rotation compensation”. The “velocity” notation has been adopted to denote that this rotation compensation term feeds the “velocity” rate equation (in contrast with a “position rotation compensation” term to be discussed in Section 7.3.3 that feeds the position rate equation). With these definitions, a comparison between (7.2.2.2-6) and (7.2.2.2-22) identifies the integral term in Equations (7.2.2.2-6) as representing the composite of sculling and velocity rotation compensation effects. Using this terminology, Equation (7.2.2.2-22) can be rewritten as:

$$\Delta \underline{v}_{SF_m}^{B_{I(m-1)}} = \underline{\upsilon}_m + \Delta \underline{v}_{Rot_m} + \Delta \underline{v}_{Scul_m} \tag{7.2.2.2-23}$$

with

$$\Delta \underline{v}_{Scul}(t) = \int_{t_{m-1}}^{t} \frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) d\tau \qquad \Delta \underline{v}_{Scul_m} = \Delta \underline{v}_{Scul}(t_m)$$

$$\underline{\alpha}(\tau) = \int_{t_{m-1}}^{\tau} \underline{\omega}_{IB}^{B} \, dt \qquad \underline{\alpha}_m = \underline{\alpha}(t_m) \tag{7.2.2.2-24}$$

$$\underline{\upsilon}(\tau) = \int_{t_{m-1}}^{\tau} \underline{a}_{SF}^{B} \, dt \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)$$

and

$$\Delta \underline{v}_{Rot_m} = \frac{1}{2}\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) \tag{7.2.2.2-25}$$

where

$\Delta \underline{v}_{Rot_m}$ = “Velocity Rotation Compensation” term.

$\Delta \underline{v}_{Scul_m}$ = “Sculling” term.

Alternatively, beginning from the (7.2.2.2-6) version:

$$\Delta \underline{v}_{SF_m}^{B_{I(m-1)}} = \underline{\upsilon}_m + \Delta \underline{v}_{Rot/Scul_m} \tag{7.2.2.2-26}$$

with

$$\Delta \underline{v}_{Rot/Scul}(t) = \int_{t_{m-1}}^{t} \left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B}\right) d\tau \qquad \Delta \underline{v}_{Rot/Scul_m} = \Delta \underline{v}_{Rot/Scul}(t_m)$$

$$\underline{\alpha}(\tau) = \int_{t_{m-1}}^{\tau} \underline{\omega}_{IB}^{B} \, dt \tag{7.2.2.2-27}$$

$$\underline{\upsilon}(\tau) = \int_{t_{m-1}}^{\tau} \underline{a}_{SF}^{B} \, dt \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)$$

where

$\Delta \underline{v}_{Rot/Scul_m}$ = Composite "Sculling" and "Velocity Rotation Compensation" term.

Equations (7.2.2.2-23) - (7.2.2.2-25) are equivalent to Equations (7.2.2.2-26) - (7.2.2.2-27); both equation sets exhibit only first order accuracy. However, Equation (7.2.2.2-23) is now in a form that enables us to substitute an expanded expression for the (7.2.2.2-25) rotation compensation term that makes (7.2.2.2-23) exact under conditions when the angular rate and specific force vectors are not rotating in the B Frame, and the ratio of magnitudes of the angular rate and specific force vectors is constant. This is an important extension because general motion is typically dominated by low frequency angular rate and specific force components that may have large amplitudes under extreme maneuvers (when second order algorithm errors may not be negligible). The extension to exactness is not possible for the (7.2.2.2-26) - (7.2.2.2-27) form because the rotation compensation effect is imbedded within the integral which includes the first order sculling term. The following sections derive an exact $\Delta \underline{v}_{Rot_m}$ velocity rotation compensation algorithm for Equation (7.2.2.2-23) in addition to digital integration algorithms for the (7.2.2.2-24) and (7.2.2.2-27) integral terms.

7.2.2.2.1 Exact Velocity Rotation Compensation Algorithm

The "exact" velocity rotation compensation algorithm is defined as the algorithm that when substituted for $\Delta \underline{v}_{Rot_m}$ in Equations (7.2.2.2-23), provides an exact solution for $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ in Equations (7.2.2-3) under conditions when the angular rate and specific force vectors are not rotating in the B Frame, and the ratio of magnitudes of the angular rate and specific force vectors is constant. The "exact" condition covers the case when the B Frame angular rate and specific force vectors are both constant, but also covers the broader case when the angular-rate/specific-force vectors have constant B Frame direction with identical time function magnitudes (e.g., in-phase sinusoidal oscillations of the angular rate and specific force magnitudes). The exact rotation compensation algorithm is derived from Equation (7.2.2-3) using (7.2.2.2-1) for $C_{B(t)}^{B_{I(m-1)}}$ when the direction of the angular rate vector is constant (i.e., the angular rate vector is not rotating which is defined in Section 7.1.1.1 as a non-coning environment). Recall from Section 7.1.1.1 (See Equation (7.1.1.1-14)) that for the case when $\underline{\omega}_{IB}^{B}$ is not rotating, $\underline{\phi}(t)$ is equal to $\underline{\alpha}(t)$ (the integral of $\underline{\omega}_{IB}^{B}$). Under this restriction, (7.2.2.2-7) applies for $\underline{\alpha}(t)$, and (7.2.2.2-1) with (7.2.2.2-7) for $\underline{\phi}(t) = \underline{\alpha}(t)$ substituted in Equation (7.2.2-3) gives:

For A Non-Coning Angular Rate Condition:

$$\Delta\underline{v}_{SF_m}^{B_{I(m-1)}} = \int_{t_{m-1}}^{t_m} \left[I + \frac{\sin\,\alpha(t)}{\alpha(t)} \left(\underline{\alpha}(t)\,\times\right) + \frac{\left(1 - \cos\,\alpha(t)\right)}{\alpha(t)^{\,2}} \left(\underline{\alpha}(t)\,\times\right)^2 \right] \underline{a}_{SF}^{B}\, dt$$

$$= \int_{t_{m-1}}^{t_m} \left[I + \sin\,\alpha(t)\,\left(\underline{u}_{\omega}\times\right) + \left(1 - \cos\,\alpha(t)\right)\left(\underline{u}_{\omega}\times\right)^2 \right] \underline{a}_{SF}^{B}\, dt \qquad (7.2.2.2.1\text{-}1)$$

For constant B Frame acceleration direction with magnitude proportional to the angular rate magnitude, Equation (7.2.2.2-10) applies for $\underline{a}_{SF}^{B}$, which when substituted in (7.2.2.2.1-1) finds:

For Non-Rotating B Frame Angular-Rate/Specific-Force Vectors With
Constant Angular-Rate/Specific-Force Magnitude Ratio:

$$\Delta\underline{v}_{SF_m}^{B_{I(m-1)}} = \int_{t_{m-1}}^{t_m} \underline{a}_{SF}^{B}\, dt + \left(\underline{u}_{\omega} \times \underline{u}_a\right) K \int_{t_{m-1}}^{t_m} \sin\,\alpha(t)\;\; \omega\, dt$$

$$+ \left[\, \underline{u}_{\omega} \times \left(\underline{u}_{\omega} \times \underline{u}_a\right)\right] K \int_{t_{m-1}}^{t_m} \left(1 - \cos\,\alpha(t)\right) \omega\, dt \qquad (7.2.2.2.1\text{-}2)$$

Substituting (7.2.2.2-6), (7.2.2.2-9) and (7.2.2.2-14) from Section 7.2.2.2 into (7.2.2.2.1-2) yields:

For Non-Rotating B Frame Angular-Rate/Specific-Force Vectors With
Constant Angular-Rate/Specific-Force Magnitude Ratio:

$$\Delta\underline{v}_{SF_m}^{B_{I(m-1)}} = \underline{\upsilon}_m + \frac{\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right)}{\alpha_m^2} \int_{\alpha = 0}^{\alpha_m} \sin\,\alpha\; d\alpha$$

$$+ \frac{\left[\underline{\alpha}_m \times \left(\underline{\alpha}_m \times \underline{\upsilon}_m\right)\right]}{\alpha_m^3} \int_{\alpha = 0}^{\alpha_m} \left(1 - \cos\,\alpha\right) d\alpha \qquad (7.2.2.2.1\text{-}3)$$

The integral terms in (7.2.2.2.1-3) are easily evaluated analytically to be:

$$\int_{\alpha = 0}^{\alpha_m} \sin\,\alpha\, d\alpha = 1 - \cos\,\alpha_m \qquad\qquad \int_{\alpha = 0}^{\alpha_m} \left(1 - \cos\,\alpha\right) d\alpha = \alpha_m - \sin\,\alpha_m \qquad (7.2.2.2.1\text{-}4)$$

Substitution in (7.2.2.2.1-3) then obtains the desired "exact" form for $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$:

For Non-Rotating B Frame Angular-Rate/Specific-Force Vectors With Constant Angular-Rate/Specific-Force Magnitude Ratio:

$$\Delta \underline{v}_{SF_m}^{B_{I(m-1)}} = \underline{\upsilon}_m + \frac{(1 - \cos \alpha_m)}{\alpha_m^2}\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) + \frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right)\left[\underline{\alpha}_m \times \left(\underline{\alpha}_m \times \underline{\upsilon}_m\right)\right] \qquad (7.2.2.2.1\text{-}5)$$

Equation (7.2.2.2.1-5) constitutes an exact solution for $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ under non-rotating B Frame angular-rate/specific-force vectors and fixed ratio between the angular-rate/specific-force magnitudes. We are now in a position to compare Equation (7.2.2.2.1-5) with Equation (7.2.2.2-23) under the "exact" conditions to identify the "exact" rotation compensation term. The discussion following Equation (7.2.2.2-22) in Section 7.2.2.2 showed that the sculling term $\Delta \underline{v}_{Scul_m}$ is zero under the "exact" conditions. Setting $\Delta \underline{v}_{Scul_m}$ to zero in (7.2.2.2-23) finds that $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ is given by:

For Non-Rotating B Frame Angular-Rate/Specific-Force Vectors With Constant Angular-Rate/Specific-Force Magnitude Ratio:

$$\Delta \underline{v}_{SF_m}^{B_{I(m-1)}} = \underline{\upsilon}_m + \Delta \underline{v}_{Rot_m} \qquad (7.2.2.2.1\text{-}6)$$

Comparing Equations (7.2.2.2.1-5) and (7.2.2.2.1-6) it should be clear that the "exact" rotation compensation term $\Delta \underline{v}_{Rot_m}$ is:

$$\Delta \underline{v}_{Rot_m} = \frac{\left(1 - \cos \alpha_m\right)}{\alpha_m^2}\, \underline{\alpha}_m \times \underline{\upsilon}_m + \frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) \underline{\alpha}_m \times \left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) \qquad (7.2.2.2.1\text{-}7)$$

The trigonometric coefficients in (7.2.2.2.1-7) can be calculated from the Taylor series expansion formulas:

$$\frac{\left(1 - \cos \alpha_m\right)}{\alpha_m^2} = \frac{1}{2\,!} - \frac{\alpha_m^2}{4\,!} + \frac{\alpha_m^4}{6\,!} - \cdots \qquad \frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) = \frac{1}{3\,!} - \frac{\alpha_m^2}{5\,!} + \frac{\alpha_m^4}{7\,!} - \cdots \qquad (7.2.2.2.1\text{-}8)$$

Equation (7.2.2.2.1-7) with (7.2.2.2.1-8) constitutes an alternative algorithm for the (7.2.2.2-25) $\Delta \underline{v}_{Rot_m}$ rotation compensation term that will generate an exact solution for $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$ in (7.2.2.2-23) under non-rotating B Frame angular-rate/specific-force vectors with fixed ratio between the angular-rate/specific-force magnitudes. In contrast, the (7.2.2.2-25)

$\Delta \underline{v}_{Rot_m}$ algorithm is accurate to only first order. Note that to first order in $\underline{\alpha}_m$, Equation (7.2.2.2.1-7) with (7.2.2.2.1-8) reduces to the Equation (7.2.2.2-25) $\Delta \underline{v}_{Rot_m}$ form (as it should).

Finally, it is to be noted that Chapter 19 (Section 19.1) describes a unified approach to strapdown algorithm design that uses a velocity translation vector (analogous to the rotation vector) for an exact calculation of $\Delta \underline{v}_{SF_m}^{B_{I(m-1)}}$. The velocity translation vector concept is formulated as an extension of Equation (7.2.2.2.1-5) to general motion (i.e., without requiring the restriction of non-rotating B Frame angular-rate/specific-force vectors with fixed ratio between the angular-rate/specific-force magnitudes invoked in this section). The unified approach was developed by the author following the original publication of this book in 2000.

7.2.2.2.2 Integrated Rate, Acceleration And Sculling Algorithm Forms

In this section we develop digital algorithms for calculating the $\underline{\upsilon}_m$, $\Delta \underline{v}_{Scul_m}$, $\Delta \underline{v}_{Rot/Scul_m}$ integral terms in Equations (7.2.2.2-23) - (7.2.2.2-24) and (7.2.2.2-26) - (7.2.2.2-27) (the $\underline{\alpha}_m$ term is provided from attitude algorithm Equations (7.1.1.1.1-17)).

We seek algorithms for calculating $\Delta \underline{v}_{Scul_m}$ and $\Delta \underline{v}_{Rot/Scul_m}$ as defined in Equations (7.2.2.2-24) and (7.2.2.2-27):

$$\Delta \underline{v}_{Scul}(t) = \int_{t_{m-1}}^{t} \frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) d\tau \qquad \Delta \underline{v}_{Scul_m} = \Delta \underline{v}_{Scul}(t_m) \qquad (7.2.2.2.2\text{-}1)$$

$$\Delta \underline{v}_{Rot/Scul}(t) = \int_{t_{m-1}}^{t} \left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B}\right) d\tau \qquad \Delta \underline{v}_{Rot/Scul_m} = \Delta \underline{v}_{Rot/Scul}(t_m) \qquad (7.2.2.2.2\text{-}2)$$

Since it is more general, and includes the (7.2.2.2.2-2) integrand, let's begin with the development of a digital algorithm for calculating the (7.2.2.2.2-1) sculling term $\Delta \underline{v}_{Scul_m}$. Following the identical procedure used in Section 7.1.1.1.1 for the coning algorithm, let us consider the integration of (7.2.2.2.2-1) as divided into portions up to and after a general time t_{l-1} within the t_{m-1} to t_m interval so that the (7.2.2.2.2-1) integral is equivalently:

$$\begin{aligned} \Delta \underline{v}_{Scul}(t) &= \Delta \underline{v}_{Scul_{l-1}} + \delta \underline{v}_{Scul}(t) \\ \delta \underline{v}_{Scul}(t) &= \int_{t_{l-1}}^{t} \frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) d\tau \end{aligned} \qquad (7.2.2.2.2\text{-}3)$$

We now define the next l cycle time point t_l within the t_{m-1} to t_m interval so that at t_l Equations (7.2.2.2.2-3) with $\underline{\alpha}(\tau)$ and $\underline{\upsilon}(\tau)$ from (7.2.2.2-24), including initial conditions, become:

$$\underline{\alpha}(\tau) = \underline{\alpha}_{l\text{-}1} + \Delta\underline{\alpha}(\tau)$$

$$\Delta\underline{\alpha}(\tau) = \int_{t_{l-1}}^{\tau} \underline{\omega}_{IB}^{B}\, dt \qquad \Delta\underline{\alpha}_l = \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}^{B}\, dt$$

$$\underline{\alpha}_l = \underline{\alpha}_{l\text{-}1} + \Delta\underline{\alpha}_l \qquad \underline{\alpha}_m = \underline{\alpha}_l(t_l = t_m) \qquad \underline{\alpha}_l = 0 \quad \text{At} \;\; \tau = t_{m\text{-}1}.$$

$$\underline{\upsilon}(\tau) = \underline{\upsilon}_{l\text{-}1} + \Delta\underline{\upsilon}(\tau)$$

$$\Delta\underline{\upsilon}(\tau) = \int_{t_{l-1}}^{\tau} \underline{a}_{SF}^{B}\, dt \qquad \Delta\underline{\upsilon}_l = \int_{t_{l-1}}^{t_l} \underline{a}_{SF}^{B}\, dt \tag{7.2.2.2.2-4}$$

$$\underline{\upsilon}_l = \underline{\upsilon}_{l\text{-}1} + \Delta\underline{\upsilon}_l \qquad \underline{\upsilon}_m = \underline{\upsilon}_l(t_l = t_m) \qquad \underline{\upsilon}_l = 0 \quad \text{At} \;\; \tau = t_{m\text{-}1}.$$

$$\Delta\underline{v}_{Scul_l} = \Delta\underline{v}_{Scul_{l\text{-}1}} + \delta\underline{v}_{Scul_l}$$

$$\delta\underline{v}_{Scul}(t) = \int_{t_{l-1}}^{t} \frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) d\tau \qquad \delta\underline{v}_{Scul_l} = \delta\underline{v}_{Scul}(t_l)$$

$$\Delta\underline{v}_{Scul_m} = \Delta\underline{v}_{Scul_l}(t_l = t_m) \qquad \Delta\underline{v}_{Scul_l} = 0 \quad \text{At} \;\; t = t_{m\text{-}1}$$

Equations (7.2.2.2.2-4) constitute the construct of a digital recursive algorithm at the l computer cycle rate for calculating the $\Delta\underline{v}_{Scul_m}$ sculling term as a summation of changes in $\Delta\underline{v}_{Scul}$ over the t_{m-1} to t_m interval. It remains to determine a digital equivalent for the $\delta\underline{v}_{Scul_l}$ integral term in (7.2.2.2.2-4).

Beginning from (7.2.2.2.2-4), we substitute $\underline{\alpha}(\tau)$ and $\underline{\upsilon}(\tau)$ into $\delta\underline{v}_{Scul}(t)$, incorporate the definitions for $\Delta\underline{\alpha}_l$ and $\Delta\underline{\upsilon}_l$, and evaluate at t_l. The result is:

$$\delta\underline{v}_{Scul_l} = \frac{1}{2}\left(\underline{\alpha}_{l\text{-}1} \times \Delta\underline{\upsilon}_l + \underline{\upsilon}_{l\text{-}1} \times \Delta\underline{\alpha}_l\right) + \int_{t_{l-1}}^{t_l} \frac{1}{2}\left(\Delta\underline{\alpha}(t) \times \underline{a}_{SF}^{B} + \Delta\underline{\upsilon}(t) \times \underline{\omega}_{IB}^{B}\right) dt \tag{7.2.2.2.2-5}$$

As for the Section 7.1.1.1.1 coning algorithm design process, development of a digital algorithm for the integral term in sculling Equation (7.2.2.2.2-5) is based on an assumed form for the B Frame angular-rate/specific-force history during the t_{l-1} to t_l time interval. Unlike the coning algorithm, very little published work exists regarding approaches for selecting angular-rate/linear-acceleration time histories for application to sculling algorithm design. In principle, the approaches used for the coning algorithm can also be applied for sculling, including optimization for sculling type motion (See discussion following Equation (7.1.1.1.1-6), and Reference 13). For this section, we provide an example based on general linearly changing angular-rate/specific-force over the t_{l-1} to t_l time interval, the coefficients for which are calculated from current and past l cycle sensor samples (as in Section 7.1.1.1.1). Thus, we approximate:

$$\underline{\omega}_{IB}^{B} \approx \underline{A} + \underline{B}\left(t - t_{l-1}\right) \qquad \underline{a}_{SF}^{B} \approx \underline{C} + \underline{D}\left(t - t_{l-1}\right) \tag{7.2.2.2.2-6}$$

where

$\underline{A}, \underline{B}, \underline{C}, \underline{D}$ = Constant vectors.

With (7.2.2.2.2-5) and the $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ definitions in (7.2.2.2.2-4):

$$\begin{aligned} \Delta\underline{\alpha}(t) &= \underline{A}\left(t - t_{l-1}\right) + \frac{1}{2}\,\underline{B}\left(t - t_{l-1}\right)^2 \\ \Delta\underline{\upsilon}(t) &= \underline{C}\left(t - t_{l-1}\right) + \frac{1}{2}\,\underline{D}\left(t - t_{l-1}\right)^2 \end{aligned} \tag{7.2.2.2.2-7}$$

Substituting (7.2.2.2.2-6) and (7.2.2.2.2-7) then yields for the integrand in (7.2.2.2.2-5):

$$\begin{aligned} \underline{\alpha}(t) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(t) \times \underline{\omega}_{IB}^{B} &= \left[\underline{A}(t - t_{l-1}) + \frac{1}{2}\,\underline{B}\,(t - t_{l-1})^2\right] \times \left[\underline{C} + \underline{D}(t - t_{l-1})\right] \\ &\quad + \left[\underline{C}(t - t_{l-1}) + \frac{1}{2}\,\underline{D}\,(t - t_{l-1})^2\right] \times \left[\underline{A} + \underline{B}(t - t_{l-1})\right] \\ &= (\underline{A} \times \underline{C})\,(t - t_{l-1}) + \left[(\underline{A} \times \underline{D}) + \frac{1}{2}\,(\underline{B} \times \underline{C})\right](t - t_{l-1})^2 + \frac{1}{2}\,(\underline{B} \times \underline{D})\,(t - t_{l-1})^3 \\ &\quad + (\underline{C} \times \underline{A})\,(t - t_{l-1}) + \left[(\underline{C} \times \underline{B}) + \frac{1}{2}\,(\underline{D} \times \underline{A})\right](t - t_{l-1})^2 + \frac{1}{2}\,(\underline{D} \times \underline{B})\,(t - t_{l-1})^3 \\ &= \frac{1}{2}\left(\underline{A} \times \underline{D} + \underline{C} \times \underline{B}\right)(t - t_{l-1})^2 \end{aligned} \tag{7.2.2.2.2-8}$$

With (7.2.2.2.2-8), the integral term in (7.2.2.2.2-5) becomes:

$$\int_{t_{l-1}}^{t_l} \frac{1}{2}\left(\Delta\underline{\alpha}(t) \times \underline{a}_{SF}^{B} + \Delta\underline{\upsilon}(t) \times \underline{\omega}_{IB}^{B}\right) dt = \frac{1}{12}\left(\underline{A} \times \underline{D} + \underline{C} \times \underline{B}\right) T_l^3 \tag{7.2.2.2.2-9}$$

where

T_l = Time interval $t_l - t_{l-1}$ (i.e., the l cycle computation period).

Following the same procedure as in Section 7.1.1.1.1, the set of $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$ constants in (7.2.2.2.2-9) can be calculated for each t_{l-1} to t_l time interval using successive measurements of integrated angular rate and specific force acceleration increments from the inertial sensors. Two such intervals would be required to uniquely determine the four constant vectors $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$ for the (7.2.2.2.2-6) linearly ramping model (a parabolic model would be characterized by 6 constant vectors and require three sensor measurements for determination, etc.). The result is then substituted for $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$ in (7.2.2.2.2-9) to derive the algorithm equivalent to the (7.2.2.2.2-5) integral term over the t_{l-1} to t_l time interval. The successive sensor integral increments can be sampled at the l cycle rate in which case measurements would be taken at t_{l-1} and t_l, each spanning t_{l-2} to t_{l-1} and $t_{l-1}$to t_l (or t_{l-2} to t_l, overall). Alternatively (as in References 11, 12 and 22), the sensor samples can be taken within the t_{l-1} to t_l time interval, two samples per l cycle for the (7.2.2.2.2-6) linearly ramping model, three for a parabolic model, etc. For sensor samples taken at the l cycle rate as in Section 7.1.1.1.1, the results for the $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$ terms in (7.2.2.2.2-9) are:

$$\begin{aligned} \underline{A} &= \frac{1}{2\,T_l}\left(\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) \qquad & \underline{B} &= \frac{1}{T_l^2}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right) \\ \underline{C} &= \frac{1}{2\,T_l}\left(\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right) \qquad & \underline{D} &= \frac{1}{T_l^2}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) \end{aligned} \tag{7.2.2.2.2-10}$$

Substitution of (7.2.2.2.2-10) in (7.2.2.2.2-9) then yields:

$$\begin{aligned} &\frac{1}{2}\int_{t_{l-1}}^{t_l}\left(\Delta\underline{\alpha}(t)\times\underline{a}_{SF}^{B} + \Delta\underline{\upsilon}(t)\times\underline{\omega}_{IB}^{B}\right)dt \\ &\quad = \frac{1}{12}\left[\frac{1}{2}\left(\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right)\times\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) + \left(\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right)\times\frac{1}{2}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \\ &\quad = \frac{1}{24}\left(\Delta\underline{\alpha}_l\times\Delta\underline{\upsilon}_l - \Delta\underline{\alpha}_l\times\Delta\underline{\upsilon}_{l-1} + \Delta\underline{\alpha}_{l-1}\times\Delta\underline{\upsilon}_l - \Delta\underline{\alpha}_{l-1}\times\Delta\underline{\upsilon}_{l-1}\right. \\ &\qquad\left. + \Delta\underline{\upsilon}_l\times\Delta\underline{\alpha}_l - \Delta\underline{\upsilon}_l\times\Delta\underline{\alpha}_{l-1} + \Delta\underline{\upsilon}_{l-1}\times\Delta\underline{\alpha}_l - \Delta\underline{\upsilon}_{l-1}\times\Delta\underline{\alpha}_{l-1}\right) \\ &\qquad\qquad = \frac{1}{12}\left(\Delta\underline{\alpha}_{l-1}\times\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\times\Delta\underline{\alpha}_l\right) \end{aligned} \tag{7.2.2.2.2-11}$$

Finally, we substitute (7.2.2.2.2-11) into (7.2.2.2.2-5) to obtain the desired equation for $\delta\underline{v}_{Scul_l}$:

$$\delta\underline{v}_{Scul_l} = \frac{1}{2}\left[\left(\underline{\alpha}_{l-1} + \frac{1}{6}\,\Delta\underline{\alpha}_{l-1}\right)\times\Delta\underline{\upsilon}_l + \left(\underline{\upsilon}_{l-1} + \frac{1}{6}\,\Delta\underline{\upsilon}_{l-1}\right)\times\Delta\underline{\alpha}_l\right] \tag{7.2.2.2.2-12}$$

A digital algorithm for $\underline{\alpha}_m$, $\underline{\upsilon}_m$ and $\Delta\underline{v}_{Scul_m}$ is determined from the above results as a composite of Equations (7.2.2.2.2-4) and (7.2.2.2.2-12):

$$\Delta\underline{\alpha}_l,\ \underline{\alpha}_l \ = \ \text{Integrated Angular Rate Sensor Outputs From Algorithm Equations (7.1.1.1.1-17)} \qquad (7.2.2.2.2\text{-}13)$$

$$\Delta\underline{\upsilon}_l \ = \ \int_{t_{l-1}}^{t_l} d\underline{\upsilon} \qquad \text{Summation Of Integrated Specific Force Output Increments From Accelerometers}$$

$$\underline{\upsilon}_l \ = \ \underline{\upsilon}_{l-1} + \Delta\underline{\upsilon}_l \qquad \underline{\upsilon}_m \ = \ \underline{\upsilon}_l(t_l = t_m) \qquad \underline{\upsilon}_l \ = \ 0 \quad \text{At } t = t_{m-1}. \qquad (7.2.2.2.2\text{-}14)$$

$$\delta\underline{v}_{\mathrm{Scul}_l} \ = \ \frac{1}{2}\left[\left(\underline{\alpha}_{l\text{-}1} + \frac{1}{6}\,\Delta\underline{\alpha}_{l\text{-}1}\right) \times \Delta\underline{\upsilon}_l + \left(\underline{\upsilon}_{l\text{-}1} + \frac{1}{6}\,\Delta\underline{\upsilon}_{l\text{-}1}\right) \times \Delta\underline{\alpha}_l\right]$$

$$\Delta\underline{v}_{\mathrm{Scul}_l} \ = \ \Delta\underline{v}_{\mathrm{Scul}_{l\text{-}1}} + \delta\underline{v}_{\mathrm{Scul}_l} \qquad \Delta\underline{v}_{\mathrm{Scul}_m} \ = \ \Delta\underline{v}_{\mathrm{Scul}_l}(t_l = t_m) \qquad (7.2.2.2.2\text{-}15)$$

$$\Delta\underline{v}_{\mathrm{Scul}_l} \ = \ 0 \quad \text{At } t = t_{m\text{-}1}$$

where

$d\underline{\upsilon} \ = \ \underline{a}_{SF}^{B}\,dt$ = Differential integrated specific force acceleration increment (i.e., analytical representation of pulse output from strapdown accelerometers).

Equations (7.2.2.2.2-15) for $\Delta\underline{v}_{\mathrm{Scul}_m}$ has been classified as a "second order algorithm" because it includes current and past l cycle $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ products. From the analysis leading to Equation (7.2.2.2.2-12), the l, l-1 cycle $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ product terms in $\delta\underline{v}_{\mathrm{Scul}_l}$ (i.e., the 1 / 6 terms) stem from the approximation of linearly ramping angular rate and specific force acceleration in the t_{l-2} to t_l time interval. If the angular rate and specific force acceleration terms were approximated as parabolically varying functions of time, a "third order algorithm" would result containing l, l-1, and l-2 cycle $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ products. If the angular rate and specific force acceleration were approximated as constants over t_{l-1} to t_l, the 1 / 6 terms in (7.2.2.2.2-15) would vanish, resulting in a "first order algorithm" for $\Delta\underline{v}_{\mathrm{Scul}_m}$. If angular rates and accelerations are slowly varying we can approximate $\Delta\underline{v}_{\mathrm{Scul}_m}$ as being equal to zero. Alternatively (and more accurately), we can set the l cycle rate equal to the m cycle rate which equates $\Delta\underline{v}_{\mathrm{Scul}_m}$ to $\delta\underline{v}_{\mathrm{Scul}_l}$ in Equations (7.2.2.2.2-15) calculated once at time t_m (and noting from the initial condition definitions in (7.2.2.2.2-13) and (7.2.2.2.2-14), that $\underline{\alpha}_{l-1}$ and $\underline{\upsilon}_{l-1}$ would be zero). Finally, we note, as in Section 7.1.1.1.1, that setting the l and m rates equal can also be achieved by increasing the m rate to match the l rate. The result would be a single high speed higher order algorithm with a simpler software architecture than the two-speed approach, but requiring more throughput. Continuing advances in the speed of modern day computers may make this the preferred approach for the future.

Using the same procedure as above for $\Delta \underline{v}_{Scul_m}$, a digital integration algorithm can also be developed for $\Delta \underline{v}_{Rot/Scul_m}$ defined in Equation (7.2.2.2.2-2). As in (7.2.2.2.2-3) - (7.2.2.2.2-4) we first obtain:

$$\underline{\alpha}(\tau) = \underline{\alpha}_{l-1} + \Delta\underline{\alpha}(\tau)$$

$$\Delta\underline{\alpha}(\tau) = \int_{t_{l-1}}^{\tau} \underline{\omega}_{IB}^{B}\, dt \qquad \Delta\underline{\alpha}_l = \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}^{B}\, dt$$

$$\underline{\alpha}_l = \underline{\alpha}_{l-1} + \Delta\underline{\alpha}_l \qquad \underline{\alpha}_l = 0 \quad \text{At} \;\; \tau = t_{m-1}$$

$$\Delta \underline{v}_{Rot/Scul_l} = \Delta \underline{v}_{Rot/Scul_{l-1}} + \delta \underline{v}_{Rot/Scul_l} \tag{7.2.2.2.2-16}$$

$$\delta \underline{v}_{Rot/Scul}(t) = \int_{t_{l-1}}^{t} \left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B}\right) d\tau \qquad \delta \underline{v}_{Rot/Scul_l} = \delta \underline{v}_{Rot/Scul}(t_l)$$

$$\Delta \underline{v}_{Rot/Scul_m} = \Delta \underline{v}_{Rot/Scul_l}(t_l = t_m)$$

$$\underline{v}_{Rot/Scul_l} = 0 \quad \text{At} \;\; t = t_{m-1}$$

Proceeding from (7.2.2.2.2-16), we substitute $\underline{\alpha}(\tau)$ in $\delta \underline{v}_{Rot/Scul}(t)$ and evaluate at t_l:

$$\delta \underline{v}_{Rot/Scul_l} = \underline{\alpha}_{l-1} \times \Delta\underline{\upsilon}_l + \int_{t_{l-1}}^{t_l} \left(\Delta\underline{\alpha}(t) \times \underline{a}_{SF}^{B}\right) dt$$

$$\Delta\underline{\upsilon}_l = \int_{t_{l-1}}^{t_l} \underline{a}_{SF}^{B}\, dt \tag{7.2.2.2.2-17}$$

With (7.2.2.2.2-6) and (7.2.2.2.2-7) for $\Delta\underline{\alpha}(t)$ and $\underline{a}_{SF}^{B}$, the integrand in (7.2.2.2.2-17) becomes:

$$\Delta\underline{\alpha}(t) \times \underline{a}_{SF}^{B} = \left[\underline{A}(t - t_{l-1}) + \frac{1}{2}\underline{B}\,(t - t_{l-1})^2\right] \times \left[\underline{C} + \underline{D}(t - t_{l-1})\right]$$

$$= (\underline{A} \times \underline{C})\,(t - t_{l-1}) + \left[(\underline{A} \times \underline{D}) + \frac{1}{2}(\underline{B} \times \underline{C})\right](t - t_{l-1})^2 + \frac{1}{2}(\underline{B} \times \underline{D})\,(t - t_{l-1})^3 \tag{7.2.2.2.2-18}$$

With (7.2.2.2.2-18), the integral term in (7.2.2.2.2-17) becomes:

$$\int_{t_{l-1}}^{t_l} \left(\Delta\underline{\alpha}(t) \times \underline{a}_{SF}^{B}\right) dt = \frac{1}{2}\,(\underline{A} \times \underline{C})\; T_l^2 + \frac{1}{3}\left[(\underline{A} \times \underline{D}) + \frac{1}{2}\,(\underline{B} \times \underline{C})\right] T_l^3 + \frac{1}{8}\,(\underline{B} \times \underline{D})\; T_l^4 \qquad (7.2.2.2.2\text{-}19)$$

Substituting (7.2.2.2.2-10) for $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$, expansion and recombination obtains:

$$\int_{t_{l-1}}^{t_l} \left(\Delta\underline{\alpha}(t) \times \underline{a}_{SF}^{B}\right) dt = \frac{1}{2}\,(\Delta\underline{\alpha}_l \times \Delta\underline{\upsilon}_l) + \frac{1}{12}\left(\Delta\underline{\alpha}_{l-1} \times \Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1} \times \Delta\underline{\alpha}_l\right) \qquad (7.2.2.2.2\text{-}20)$$

With (7.2.2.2.2-20), the $\delta\underline{v}_{Rot/Scul_l}$ term in Equations (7.2.2.2.2-17) then becomes:

$$\delta\underline{v}_{Rot/Scul_l} = \left(\underline{\alpha}_{l-1} + \frac{1}{2}\,\Delta\underline{\alpha}_l\right) \times \Delta\underline{\upsilon}_l + \frac{1}{12}\left(\Delta\underline{\alpha}_{l-1} \times \Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1} \times \Delta\underline{\alpha}_l\right) \qquad (7.2.2.2.2\text{-}21)$$

A digital algorithm for $\underline{\upsilon}_m$ and $\Delta\underline{v}_{Rot/Scul_m}$ in (7.2.2.2-26) is determined from the above results as a composite of Equations (7.2.2.2.2-21) and (7.2.2.2.2-16) with $\underline{\alpha}$, $\Delta\underline{\alpha}$, $\underline{\upsilon}$, $\Delta\underline{\upsilon}$ terms from (7.2.2.2.2-13) - (7.2.2.2.2-14):

$$\Delta\underline{\alpha}_l,\ \underline{\alpha}_l = \begin{array}{l}\text{Integrated Angular Rate Sensor Outputs}\\ \text{From Algorithm Equations (7.1.1.1.1-17)}\end{array} \qquad (7.2.2.2.2\text{-}22)$$

$$\Delta\underline{\upsilon}_l = \int_{t_{l-1}}^{t_l} d\underline{\upsilon} \qquad \begin{array}{c}\text{Summation Of Integrated Specific Force}\\ \text{Output Increments From Accelerometers}\end{array}$$

$$\underline{\upsilon}_l = \underline{\upsilon}_{l-1} + \Delta\underline{\upsilon}_l \qquad \underline{\upsilon}_m = \underline{\upsilon}_l(t_l = t_m) \qquad \underline{\upsilon}_l = 0 \quad \text{At } t = t_{m-1}. \qquad (7.2.2.2.2\text{-}23)$$

$$\delta\underline{v}_{Rot/Scul_l} = \left(\underline{\alpha}_{l-1} + \frac{1}{2}\,\Delta\underline{\alpha}_l\right) \times \Delta\underline{\upsilon}_l + \frac{1}{12}\left(\Delta\underline{\alpha}_{l-1} \times \Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1} \times \Delta\underline{\alpha}_l\right)$$

$$\begin{aligned} \Delta\underline{v}_{Rot/Scul_l} &= \Delta\underline{v}_{Rot/Scul_{l-1}} + \delta\underline{v}_{Rot/Scul_l} \\ \Delta\underline{v}_{Rot/Scul_m} &= \Delta\underline{v}_{Rot/Scul_l}(t_l = t_m) \\ \Delta\underline{v}_{Rot/Scul_l} &= 0 \quad \text{At } t = t_{m-1} \end{aligned} \qquad (7.2.2.2.2\text{-}24)$$

Equation (7.2.2.2.2-24) for $\Delta\underline{v}_{Rot/Scul_m}$ has been classified as a "second order algorithm" because it includes current and past *l* cycle $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ products. From the analysis leading to Equation (7.2.2.2.2-21), the *l* and *l*-1 cycle $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ product terms in $\delta\underline{v}_{Rot/Scul_l}$ (i.e., the 1 / 12 term) stem from the approximation of linearly ramping angular rate and specific force acceleration in the t_{l-2} to t_l time interval. If the angular rate and specific force acceleration terms were approximated as parabolically varying functions of time, a "third order algorithm" would result containing l, l-1, and l-2 cycle $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ products. If the angular rate and specific force acceleration were approximated as constants (i.e., slowly varying) over t_{l-1} to t_l, the 1 / 12 term in (7.2.2.2.2-24) would vanish, resulting in a "first order algorithm" for $\delta\underline{v}_{Rot/Scul_l}$. Alternatively, for slowly varying angular rates and accelerations we can set the l cycle rate equal to the m cycle rate which equates $\Delta\underline{v}_{Rot/Scul_m}$ to $\delta\underline{v}_{Rot/Scul_l}$ in Equations (7.2.2.2.2-24) calculated once at time t_m (and noting from the initial condition definitions in (7.2.2.2.2-22) that $\underline{\alpha}_{l-1}$ would be zero). Finally, we note, as in Section 7.1.1.1.1, that setting the l and m rates equal can also be achieved by increasing the m rate to match the l rate. The result would be a single high speed higher order algorithm with a simpler software architecture than the two-speed approach, but requiring more throughput. Continuing advances in the speed of modern day computers may make this the preferred approach for the future.

7.3 POSITION UPDATE ALGORITHMS

In this section we develop digital integration algorithms for calculating position relative to the earth in the form of altitude (h) above the earth surface and the C_N^E direction cosine matrix defining the angular attitude between the local level N Frame and the earth fixed E Frame (from which latitude/longitude can be extracted). Two algorithm forms are developed; a "typical" form based on trapezoidal integration of velocity, and a high resolution form which accounts for dynamic attitude and velocity changes within the position update period. The high resolution algorithm is modeled after the Section 7.2 two-speed velocity update approach; a moderate speed algorithm is used for position update that is analytically exact under constant angular-rate/specific-force-acceleration during the position update period; a high speed algorithm is then used to measure rectifying dynamic angular-rate/specific-force effects (denoted by the writer as "scrolling") for input to the moderate speed algorithm.

Both the typical and high resolution forms can be represented by continuous differential Equations (4.4.1.2-8) and (4.4.1.1-1) repeated below:

$$\dot{h} = \underline{u}_{ZN}^N \cdot \underline{v}^N \tag{7.3-1}$$

$$\dot{C}_N^E = C_N^E \left(\underline{\rho}^N \times\right) \tag{7.3-2}$$

where

$\underline{\rho}^N$ = Transport rate (also known as $\underline{\omega}_{EN}^N$), the angular rate of the N frame relative to the E Frame, projected on N Frame axes.

The typical and high resolution position algorithm forms derive from a general updating formulation for h and C_N^E.

7.3.1 POSITION UPDATING IN GENERAL

The general altitude h updating algorithm is formulated as the integral of Equation (7.3-1) over a computer update cycle:

$$h_n = h_{n-1} + \Delta h_n \tag{7.3.1-1}$$

$$\Delta h_n = \int_{t_{n-1}}^{t_n} \underline{u}_{ZN}^N \cdot \underline{v}^N \, dt \tag{7.3.1-2}$$

where

n = Computer cycle index for position updates.

Allowing for a higher speed digital computation loop (i.e., the m loop for attitude and velocity integration), Equation (7.3.1-2) can be written as:

$$\Delta h_n = \underline{u}_{ZN}^N \cdot \sum_{m=1}^{j} \Delta \underline{R}_m^N \tag{7.3.1-3}$$

$$\Delta \underline{R}_m^N \equiv \int_{t_{m-1}}^{t_m} \underline{v}^N \, dt \tag{7.3.1-4}$$

where

j = Number of m cycles in an n cycle.

If vertical channel control is to be included as in Equations (4.4.1.2.1-1) - (4.4.1.2.1-3), the following additional altitude update equation would be included:

$$h_{n_+} = h_{n_-} - e_{vc2_n} T_n \tag{7.3.1-5}$$

where

-, + = Indicators for h_n value before (-) and after (+) the vertical stabilization addition.

e_{vc2_n} = Altitude control signal calculated as in Equations (7.2-6).

T_n = Time interval between position update cycles.

The general updating algorithm for the C_N^E direction cosine matrix is designed to achieve the same numerical result at the update times as would the formal continuous integration of the

Equations (7.3-2) $\dot{C}_N^E$ expression at the same time instant. The algorithm is constructed by envisioning the local level navigation N Frame angular history in the digital updating world (produced in Equation (7.3-2) by $\underline{\rho}^N$) as being constructed of successive discrete attitudes relative to the earth (E Frame) at each update time instant. We apply special nomenclature to describe the coordinate frame attitude history where:

$N_{E(n)}$ = Coordinate frame corresponding to the discrete attitude of the N Frame in rotating earth space (E) at computer update time t_n.

With this definition, the general updating algorithm for C_N^E is constructed as follows using the Equation (3.2.1-5) direction cosine matrix product chain rule:

$$C_{N_{E(n)}}^{E} = C_{N_{E(n-1)}}^{E} \, C_{N_{E(n)}}^{N_{E(n-1)}} \tag{7.3.1-6}$$

where

$C_{N_{E(n-1)}}^{E}$ = C_N^E relating the N Frame at time t_{n-1} to the E Frame.

$C_{N_{E(n)}}^{E}$ = C_N^E relating the N Frame at time t_n to the E Frame.

$C_{N_{E(n)}}^{N_{E(n-1)}}$ = Direction cosine matrix that accounts for N Frame rotation relative to the E Frame from its attitude at time t_{n-1} to its attitude at time t_n.

The $C_{N_{E(n)}}^{N_{E(n-1)}}$ matrix in Equation (7.3.1-6) is defined formally as:

$$C_{N_{E(n)}}^{N_{E(n-1)}} = I + \int_{t_{n-1}}^{t_n} \dot{C}_{N(t)}^{N_{E(n-1)}} \, dt \tag{7.3.1-7}$$

where

$C_{N(t)}^{N_{E(n-1)}}$ = Direction cosine matrix relating the N Frame attitude at an arbitrary time t in the interval t_{n-1} to t_n, with its $N_{E(n-1)}$ attitude.

Following the same development procedure for $C_{B_{I(m)}}^{B_{I(m-1)}}$ in Section 7.1.1.1, the $C_{N_{E(n)}}^{N_{E(n-1)}}$ matrix can also be expressed in terms of a rotation vector defining the Frame $N_{E(n)}$ attitude relative to Frame $N_{E(n-1)}$. Applying Equations (3.2.2.1-8) and (3.2.2.1-9) obtains:

$$C_{N_{E(n)}}^{N_{E(n-1)}} = I + \frac{\sin \xi_n}{\xi_n}\left(\underline{\xi}_n \times\right) + \frac{(1 - \cos \xi_n)}{\xi_n^{\,2}}\left(\underline{\xi}_n \times\right)\left(\underline{\xi}_n \times\right)$$

$$\frac{\sin \xi_n}{\xi_n} = 1 - \frac{\xi_n^{\,2}}{3\,!} + \frac{\xi_n^{\,4}}{5\,!} - \cdots \qquad \frac{(1 - \cos \xi_n)}{\xi_n^{\,2}} = \frac{1}{2\,!} - \frac{\xi_n^{\,2}}{4\,!} + \frac{\xi_n^{\,4}}{6\,!} - \cdots \tag{7.3.1-8}$$

where

$\underline{\xi}_n$ = Rotation vector defining the Frame $N_{E(n)}$ attitude at time t_n relative to the Frame $N_{E(n-1)}$ attitude at time t_{n-1}.

ξ_n = Magnitude of $\underline{\xi}_n$.

The angular rotation rate of the N Frame relative to the earth $\underline{\rho}^N$ is small and typically no larger than one or two earth rates. As such, because the t_{n-1} to t_n update cycle is relatively short, $\underline{\xi}_n$ will be very small in magnitude. Since $\underline{\rho}^N$ is small and slowly changing over a typical t_{n-1} to t_n update cycle (due to small changes in velocity and position over this time period) $\underline{\rho}^N$ can be approximated as non-rotating. The result is that $\underline{\xi}_n$ for (7.3.1-8) can be calculated as the integral of the simplified form of the Equation (3.3.5-14) rotation vector rate expression in which the cross-product terms are neglected:

$$\underline{\xi}_n \approx \int_{t_{n-1}}^{t_n} \underline{\rho}^N \, dt \tag{7.3.1-9}$$

A discrete digital algorithm for the Equation (7.3.1-9) $\underline{\xi}_n$ integral can be constructed by first approximating $\underline{\rho}^N$ from (4.1.1-6) as:

$$\underline{\rho}^N \approx \rho_{ZN_{n-1/2}}\, \underline{u}_{ZN}^N + F_{C_{n-1/2}}^N \left(\underline{u}_{ZN}^N \times \underline{v}^N\right) \tag{7.3.1-10}$$

where

n-1/2 = Subscript indicating value for parameter midway between times t_{n-1} and t_n.

F_C^N = Curvature matrix (3 by 3) in the N Frame that is a function of position (C_N^E, h) with elements 3, i and i, 3 equal to zero and the remaining elements symmetrical about the diagonal. For a spherical earth model, the “remaining” elements are zero off the diagonal and the reciprocal of the radial distance from earth center to the INS on the diagonal. For an ellipsoidal earth model, the “remaining” terms represent the local curvature on the earth surface projected to the INS altitude (See Equations (5.3-18) for closed-form expression).

Using (7.3.1-10) in (7.3.1-9) and applying the (7.3.1-4) definition, then obtains:

$$\underline{\xi}_n \approx \rho_{ZN_{n-1/2}} \underline{u}_{ZN}^N T_n + F_{C_{n-1/2}}^N \left(\underline{u}_{ZN}^N \times \sum_{m=1}^{j} \Delta \underline{R}_m^N \right) \tag{7.3.1-11}$$

The $(\,)_{n-1/2}$ terms in (7.3.1-11) are all functions of position which has not yet been updated. Hence, in order to calculate the $(\,)_{n-1/2}$ terms, an approximate extrapolation formula must be used based on previously computed values for the $(\,)$ parameters. For example, a linear extrapolation formula using the last two computed values for $(\,)$ would be:

$$(\,)_{n-1/2} \approx (\,)_{n-1} + \frac{1}{2}\left[(\,)_{n-1} - (\,)_{n-2}\right] = \frac{3}{2}(\,)_{n-1} - \frac{1}{2}(\,)_{n-2} \tag{7.3.1-12}$$

The method for calculating the $\Delta \underline{R}_m^N$ term for (7.3.1-3) and (7.3.1-11) from the Equation (7.3.1-4) integral depends on whether typical (e.g., trapezoidal) integration is used for position updating or whether a more precision high resolution integration approach is to be applied.

We note in passing that based on the smallness of $\underline{\xi}_n$, Equations (7.3.1-8) for $C_{N_{E(n)}}^{N_{E(n-1)}}$ can also be simplified. For example, a second order algorithm version (accurate to second order in $\underline{\xi}_n$) is from(7.3.1-8):

$$C_{N_{E(n)}}^{N_{E(n-1)}} \approx I + \left(\underline{\xi}_n \times\right) + \frac{1}{2}\left(\underline{\xi}_n \times\right)\left(\underline{\xi}_n \times\right) \tag{7.3.1-13}$$

Except for the initial alignment argument, the comments at the end of Section (7.1.1.2) regarding the advisability of using the simplified Equation (7.1.1.2-5) direction cosine local level frame updating algorithm also apply regarding use of Equation (7.3.1-13) for $C_{N_{E(n)}}^{N_{E(n-1)}}$ rather than the complete Equation (7.3.1-8) form.

7.3.2 <u>TYPICAL POSITION UPDATING ALGORITHM</u>

Applying typical trapezoidal integration for the h and C_N^E updating process would utilize Equations (7.3.1-1), (7.3.1-3), (7.3.1-5), (7.3.1-6), (7.3.1-8), (7.3.1-11) and (7.3.1-12) with a trapezoidal integration algorithm in (7.3.1-4) for $\Delta \underline{R}_m^N$:

$$\Delta \underline{R}_m^N \approx \frac{1}{2}\left(\underline{v}_m^N + \underline{v}_{m-1}^N\right) T_m \tag{7.3.2-1}$$

7.3.3 HIGH RESOLUTION POSITION UPDATING ALGORITHMS

The high resolution approach for implementing the h and C_N^E updating process would utilize Equations (7.3.1-1), (7.3.1-3), (7.3.1-5), (7.3.1-6), (7.3.1-8), (7.3.1-11) and (7.3.1-12) with a high speed digital integration algorithm in (7.3.1-4) for $\Delta \underline{R}_m^N$.

The digital algorithm for $\Delta \underline{R}_m^N$ is developed by first expanding the (7.3.1-4) $\underline{v}^N$ integrand. Using the expression for $\underline{v}_m^N$ in Equations (7.2-2) with (7.2-4), $\underline{v}^N$ can be defined as a continuous time function at a general time point t since the last t_{m-1} update:

$$\underline{v}^N(t) = \underline{v}_{m-1}^N + C_L^N \, \Delta \underline{v}_{SF}^L(t) + \Delta \underline{v}_{G/COR_m}^N \frac{(t - t_{m-1})}{T_m}$$
$$\Delta \underline{v}_{SF}^L(t) = \int_{t_{m-1}}^{t} C_B^L \, \underline{a}_{SF}^B \, d\tau \qquad (7.3.3\text{-}1)$$

Equations (7.3.3-1) are based on the assumption that gravity/Coriolis term $\Delta \underline{v}_{G/COR_m}^N$ can be approximated as the integral of a constant over t_{m-1} to t_m. With (7.3.3-1), $\Delta \underline{R}_m^N$ from (7.3.1-4) is given by:

$$\Delta \underline{R}_m^N = \left(\underline{v}_{m-1}^N + \frac{1}{2} \Delta \underline{v}_{G/COR_m}^N \right) T_m + C_L^N \, \Delta \underline{R}_{SF_m}^L$$
$$\Delta \underline{R}_{SF_m}^L = \int_{t_{m-1}}^{t_m} \Delta \underline{v}_{SF}^L(t) \, dt \qquad \Delta \underline{v}_{SF}^L(t) = \int_{t_{m-1}}^{t} C_B^L \, \underline{a}_{SF}^B \, d\tau \qquad (7.3.3\text{-}2)$$

where

$\Delta \underline{R}_{SF_m}^L$ = L Frame coordinate portion of $\Delta \underline{R}_m$ produced by specific force acceleration.

Equations (7.2.2-2), (7.2.2-4) and (7.2.2.2-22) show that $\Delta \underline{v}_{SF}^L(t)$ in (7.3.3-2) can be approximated to first order (in body rotation angle) by:

$$\Delta \underline{v}_{SF}^L(t) = \frac{1}{2} \left(C_{L_{I(n-1)}}^{L(t)} + C_{L_{(n-1)}}^{L_{(m-1)}} \right) \Delta \underline{v}_{SF}^{L_{(n-1)}}(t) = \frac{1}{2} \left(C_{L_{(m-1)}}^{L(t)} + I \right) C_{L_{(n-1)}}^{L_{(m-1)}} \Delta \underline{v}_{SF}^{L_{(n-1)}}(t)$$
$$= \frac{1}{2} \left(C_{L_{(m-1)}}^{L(t)} - I \right) C_{L_{(n-1)}}^{L_{(m-1)}} \Delta \underline{v}_{SF}^{L_{(n-1)}}(t) + C_{L_{(n-1)}}^{L_{(m-1)}} \Delta \underline{v}_{SF}^{L_{(n-1)}}(t) \qquad (7.3.3\text{-}3)$$

(Continued)

$$= \frac{1}{2}\left(C_{L(m-1)}^{L(m)} - I\right)\frac{\left(t - t_{m-1}\right)}{T_m}\, C_{L(n-1)}^{L(m-1)}\, \Delta\underline{v}_{SF_m}^{L(n-1)}\, \frac{\left(t - t_{m-1}\right)}{T_m} + C_{L(n-1)}^{L(m-1)}\, \Delta\underline{v}_{SF}^{L(n-1)}(t)$$

$$= \frac{1}{2}\left(C_{L(m-1)}^{L(m)} - I\right) C_{L(n-1)}^{L(m-1)}\, \Delta\underline{v}_{SF_m}^{L(n-1)}\, \frac{\left(t - t_{m-1}\right)^2}{T_m^2} + C_{L(n-1)}^{L(m-1)}\, \Delta\underline{v}_{SF}^{L(n-1)}(t)$$

$$= \frac{1}{2}\left(C_{L(n-1)}^{L(m)} - C_{L(n-1)}^{L(m-1)}\right) \Delta\underline{v}_{SF_m}^{L(n-1)}\, \frac{\left(t - t_{m-1}\right)^2}{T_m^2} + C_{L(n-1)}^{L(m-1)}\, \Delta\underline{v}_{SF}^{L(n-1)}(t) \qquad \begin{array}{c}(7.3.3\text{-}3)\\ \text{(Continued)}\end{array}$$

$$\Delta\underline{v}_{SF}^{L(n-1)}(t) = C_{B(m-1)}^{L(n-1)}\, \Delta\underline{v}_{SF}^{B(m-1)}(t)$$

$$\Delta\underline{v}_{SF}^{B(m-1)}(t) = \underline{\upsilon}(t) + \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) + \int_{t_{m-1}}^{t} \frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) d\tau$$

$$\underline{\alpha}(\tau) = \int_{t_{m-1}}^{\tau} \underline{\omega}_{IB}^{B}\, dt \qquad\qquad \underline{\upsilon}(\tau) = \int_{t_{m-1}}^{\tau} \underline{a}_{SF}^{B}\, dt$$

where

$C_{B(m-1)}^{L(n-1)}$ = C_B^L matrix updated for B Frame motion at time $t = t_{m-1}$ and for L Frame motion at time $t = t_{n-1}$.

$C_{L(n-1)}^{L(m)}$, $C_{L(n-1)}^{L(m-1)}$ = Current and past m cycle values for the direction cosine matrix relating Frame L at times t_{n-1} and t_m (as calculated with Equations (7.2.2.1-1) and (7.2.2.1-4)).

In Equations (7.3.3-3), the "I" notation in subscripts and superscripts used for clarity in Equations (7.2.2-2) and (7.2.2-4), has been dropped for simplicity. In addition, $\left(C_{L(m-1)}^{L(t)} - I\right)$ and $\Delta\underline{v}_{SF}^{L(n-1)}(t)$ in the first part of the $\Delta\underline{v}_{SF}^{L}(t)$ expression have been approximated to be linearly ramping in time over t_{m-1} to t_m. Note also, as discussed in Section 7.2.2.1, that the $C_{L(n-1)}^{L}$ terms in (7.3.3-3) can be approximated by the identity matrix for all but very high precision applications.

Based on (7.3.3-3), the (7.2.2.2-24) definitions, and including the simplified (7.2.2.1-1) form for $C_{L(n-1)}^{L(m)}$, the $\Delta\underline{R}_{SF_m}^{L}$ term in (7.3.3-2) can be defined by the equivalent forms:

$$\Delta \underline{R}_{SF_m}^{L} = -\frac{1}{6}\left[\left(\underline{\zeta}_{n-1,m} - \underline{\zeta}_{n-1,m-1}\right)\times\right] \Delta \underline{v}_{SF_m}^{L(n-1)}\, T_m + C_{L(n-1)}^{L(m-1)}\, C_{B(m-1)}^{L(n-1)}\, \Delta \underline{R}_{SF_m}^{B(m-1)}$$

$$\Delta \underline{R}_{SF_m}^{B(m-1)} = \int_{t_{m-1}}^{t_m} \Delta \underline{v}_{SF}^{B(m-1)}(t)\, dt = \int_{t_{m-1}}^{t_m} \left[\underline{\upsilon}(t) + \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) + \Delta \underline{v}_{Scul}(t)\right] dt$$

$$\Delta \underline{v}_{Scul}(t) = \frac{1}{2}\int_{t_{m-1}}^{t} \left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) d\tau \qquad (7.3.3\text{-}4)$$

$$\underline{\alpha}(\tau) = \int_{t_{m-1}}^{\tau} \underline{\omega}_{IB}^{B}\, dt \qquad \underline{\upsilon}(\tau) = \int_{t_{m-1}}^{\tau} \underline{a}_{SF}^{B}\, dt$$

where

$\Delta \underline{v}_{Scul}(t)$ = Velocity change since t_{m-1} due to sculling oscillatory motion.

Following a similar development path as used in Section 7.2.2.2 for the body frame integrated specific force acceleration increment, the $\int \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) dt$ term in the Equation (7.3.3-4) $\Delta \underline{R}_{SF_m}^{B(m-1)}$ expression can be revised into a non-integral term plus an integral term that vanishes under constant angular-rate/specific-force-acceleration, both being of first order accuracy. The non-integral term will then be extended into a more accurate form that is exact under constant angular-rate/specific-force. We begin using classical integration by parts substitution (as in Section 7.2.2.2 leading to (7.2.2.2-21)), to show that the $\int \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) dt$ term has the following equivalent forms:

$$r_0 = \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) dt$$

$$r_1 = \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) dt = \frac{1}{2}\left(\underline{S}_{\alpha_m} \times \underline{\upsilon}_m\right) - \frac{1}{2}\int_{t_{m-1}}^{t_m} \left(\underline{S}_{\alpha}(t) \times \underline{a}_{SF}^{B}\right) dt$$

$$r_2 = \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) dt = \frac{1}{2}\left(\underline{\alpha}_m \times \underline{S}_{\upsilon_m}\right) + \frac{1}{2}\int_{t_{m-1}}^{t_m} \left(\underline{S}_{\upsilon}(t) \times \underline{\omega}_{IB}^{B}\right) dt \qquad (7.3.3\text{-}5)$$

(Continued)

$$\underline{S}_{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\alpha}(\tau)\, d\tau \qquad \underline{S}_{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{\upsilon}(\tau)\, d\tau \tag{7.3.3-5 (Continued)}$$

$$\underline{\alpha}_m = \underline{\alpha}(t_m) \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)$$

$$\underline{S}_{\alpha_m} = \underline{S}_{\alpha}(t_m) \qquad \underline{S}_{\upsilon_m} = \underline{S}_{\upsilon}(t_m)$$

where

r_0 = Original analytical form of the integral term.

r_1, r_2 = Alternative analytical forms of the integral term.

$\underline{S}_{\alpha}, \underline{S}_{\upsilon}$ = Time integral of $\underline{\alpha}, \underline{\upsilon}$.

Since r_1 and r_2 are analytically equivalent to the original r_0 integral form, we can write:

$$\int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) dt = \frac{1}{3}\left(r_0 + r_1 + r_2\right) \tag{7.3.3-6}$$

Substituting for r_0, r_1, and r_2 from (7.3.3-5) into (7.3.3-6) and combining terms then yields:

$$\begin{aligned}\int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\upsilon}(t)\right) dt = &\ \frac{1}{6}\left(\underline{S}_{\alpha_m} \times \underline{\upsilon}_m + \underline{\alpha}_m \times \underline{S}_{\upsilon_m}\right) \\ &- \frac{1}{6}\int_{t_{m-1}}^{t_m}\left[\underline{S}_{\alpha}(t) \times \underline{a}_{SF}^{B} - \underline{S}_{\upsilon}(t) \times \underline{\omega}_{IB}^{B} - \underline{\alpha}(t) \times \underline{\upsilon}(t)\right] dt\end{aligned} \tag{7.3.3-7}$$

We now substitute Equation (7.3.3-7) with the Equation (7.3.3-5) definitions into Equations (7.3.3-2) and (7.3.3-4) to obtain the desired form for calculating $\Delta\underline{R}_m^N$:

$$\begin{aligned}\Delta\underline{R}_m^N &= \left(\underline{v}_{m-1}^N + \frac{1}{2}\Delta\underline{v}_{G/COR_m}^N\right) T_m + C_L^N\, \Delta\underline{R}_{SF_m}^L \\ \Delta\underline{R}_{SF_m}^L &= -\frac{1}{6}\left[\left(\underline{\zeta}_{n-1,m} - \underline{\zeta}_{n-1,m-1}\right)\times\right] \Delta\underline{v}_{SF_m}^{L(n-1)}\, T_m + C_{L(n-1)}^{L(m-1)}\, C_{B(m-1)}^{L(n-1)}\, \Delta\underline{R}_{SF_m}^{B(m-1)}\end{aligned} \tag{7.3.3-8}$$

$$\Delta\underline{R}_{SF_m}^{B(m-1)} = \underline{S}_{\upsilon_m} + \Delta\underline{R}_{Rot_m} + \Delta\underline{R}_{Scrl_m} \tag{7.3.3-9}$$

with

$$\Delta \underline{R}_{Scrl_m} = \int_{t_{m-1}}^{t_m} \frac{1}{6}\left[6\, \Delta \underline{v}_{Scul}(t) - \underline{S}_{\alpha}(t) \times \underline{a}_{SF}^{B} + \underline{S}_{\upsilon}(t) \times \underline{\omega}_{IB}^{B} + \underline{\alpha}(t) \times \underline{\upsilon}(t) \right] dt$$

$$\Delta \underline{v}_{Scul}(t) = \int_{t_{m-1}}^{t} \frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B} \right) d\tau$$

$$\underline{S}_{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\alpha}(\tau)\, d\tau \qquad \underline{S}_{\alpha_m} = \underline{S}_{\alpha}(t_m) \tag{7.3.3-10}$$

$$\underline{S}_{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{\upsilon}(\tau)\, d\tau \qquad \underline{S}_{\upsilon_m} = \underline{S}_{\upsilon}(t_m)$$

$$\underline{\alpha}(\tau) = \int_{t_{m-1}}^{\tau} \underline{\omega}_{IB}^{B}\, dt \qquad \underline{\alpha}_m = \underline{\alpha}(t_m)$$

$$\underline{\upsilon}(\tau) = \int_{t_{m-1}}^{\tau} \underline{a}_{SF}^{B}\, dt \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)$$

and

$$\Delta \underline{R}_{Rot_m} = \frac{1}{6}\left(\underline{S}_{\alpha_m} \times \underline{\upsilon}_m + \underline{\alpha}_m \times \underline{S}_{\upsilon_m} \right) \tag{7.3.3-11}$$

where

$\Delta \underline{R}_{Rot_m}$ = "Position Rotation Compensation" analogous to the "Velocity Rotation Compensation" term in Equation (7.2.2.2-23).

$\Delta \underline{R}_{Scrl_m}$ = "Scrolling" term analogous to the "sculling" term in Equation (7.2.2.2-23). The term "scrolling" was coined by the author merely to have a name for the term and also to have one that sounds like "sculling", but for position integration (change in the position vector $\underline{R}$ stressing the "R" sound). The complex mathematical derivations and associated algorithms that accompany "scrolling" may prove to be a more appropriate reason for the name.

The $\Delta \underline{v}_{G/COR_m}^{N}$, $\Delta \underline{v}_{SF_m}^{L_{(n-1)}}$, $C_{L_{(n-1)}}^{L_{(m-1)}}$ and $\underline{\zeta}_{n-1,m}$ terms in (7.3.3-8) are provided from Equations (7.2.1-1) - (7.2.1-3), (7.2.2-2), (7.2.2.1-1), (7.2.2.1-4) and (7.2.2.1-6) - (7.2.2.1-7) of the velocity update computations.

A key characteristic of (7.3.3-9) is that the $\Delta \underline{R}_{Scrl_m}$ scrolling term is identically zero under constant body axis angular rate and specific force conditions. This can be readily verified from Equations (7.3.3-10) by substitution of a constant angular rate and specific force vector for the $\underline{\omega}_{IB}^{B}$ and $\underline{a}_{SF}^{B}$ terms, and carrying out the indicated operations analytically. As such, $\Delta \underline{R}_{Scrl_m}$ will only produce an output under the presence of dynamic body axis angular-rate/specific-force components. This is an important characteristic because for most real dynamic environments, the magnitude of high frequency angular-rate/specific-force is small so that first order approximations accurately apply (first order in integrated body angular-rate/specific-force over the t_{m-1} to t_m time interval, the basis for the analytical form of $\Delta \underline{R}_{Scrl_m}$ in Equations (7.3.3-10)). We conclude that the analytical form for $\Delta \underline{R}_{Scrl_m}$ will also yield a reasonably accurate solution under situations when the low frequency body angular rate and specific force components are large. In contrast, the Equation (7.3.3-11) position rotation compensation $\Delta \underline{R}_{Rot_m}$ term (also based on first order approximations) can have noticeable second order error under extreme maneuvers. However, the form of (7.3.3-9) which has $\Delta \underline{R}_{Rot_m}$ separate from other terms, allows us to expand $\Delta \underline{R}_{Rot_m}$ (in the following section) to a more accurate form that is exact under constant angular-rate/specific-force (similar to the velocity rotation compensation term in Section 7.2.2.2.1).

From the characteristics of the $\Delta \underline{R}_{Scrl_m}$ scrolling term delineated above, we see that this term is to position integration what the "coning" term is to attitude integration and what the "sculling" term is to velocity integration; each accounts for higher frequency body frame dynamic inputs (i.e., angular rate and, if applicable, specific force acceleration) within the digital integration m cycle update period; each is zero under constant body rate and, if applicable, constant body specific force acceleration conditions. The coning and sculling terms are also zero under the broader definitions of non-rotating B Frame angular rate (for coning), and non-rotating B Frame angular-rate/specific-force coupled with constant angular-rate/specific-force magnitude ratio (for sculling).

The following sections develop algorithms for the exact position rotation compensation term in (7.3.3-9), and for the scrolling and other integral terms in (7.3.3-10).

7.3.3.1 EXACT POSITION ROTATION COMPENSATION ALGORITHM

The derivation of Equations (7.3.3-8) - (7.3.3-11) was based on small-angle/velocity-change assumptions for the $\Delta \underline{R}_{Rot_m}$ and $\Delta \underline{R}_{Scrl_m}$ terms in the $\Delta \underline{R}_{SF_m}^{B(m-1)}$ expression. An improved accuracy version of $\Delta \underline{R}_{Rot_m}$ can also be developed by specifying the solution to be exact under constant body angular rate and specific force acceleration, and to first order, equal $\Delta \underline{R}_{Rot_m}$ in Equation (7.3.3-11) under general angular-rate/specific-force conditions. The improved version, when utilized in Equations (7.3.3-9) yields a more accurate integration algorithm under large magnitude angular-rate/linear-acceleration conditions than is provided by the Equation (7.3.3-11) first order accuracy form.

The derivation of the exact position rotation compensation algorithm begins by returning to the basic definition for $\Delta \underline{R}_{SF_m}^{B(m-1)}$ in Equations (7.3.3-4):

$$\Delta \underline{R}_{SF_m}^{B(m-1)} = \int_{t_{m-1}}^{t_m} \Delta \underline{v}_{SF}^{B(m-1)}(t)\, dt \tag{7.3.3.1-1}$$

Under constant B Frame specific force acceleration and non-coning angular rate, Equation (7.2.2.2.1-1) becomes for $\Delta \underline{v}_{SF}^{B_{I(m-1)}}$ at a general time t following t_{m-1}:

For Non-Coning Angular Rate And Constant B Frame Specific Force:

$$\begin{aligned} \Delta \underline{v}_{SF}^{B_{I(m-1)}}(t) = & \int_{t_{m-1}}^{t} \underline{a}_{SF}^{B}\, d\tau + \left(\underline{u}_\omega \times \underline{a}_{SF}^{B}\right) \int_{t_{m-1}}^{t} \sin\, \alpha(\tau)\, d\tau \\ & + \left[\underline{u}_\omega \times \left(\underline{u}_\omega \times \underline{a}_{SF}^{B}\right)\right] \int_{t_{m-1}}^{t} \left(1 - \cos\, \alpha(\tau)\right) d\tau \end{aligned} \tag{7.3.3.1-2}$$

where

$\underline{u}_\omega$ = Unit vector along $\underline{\omega}_{IB}^{B}$ in which $\underline{u}_\omega$ is constant in the B Frame.

$\alpha(\tau)$ = Magnitude of the integral of $\underline{\omega}_{IB}^{B}$ from time t_{m-1} to time τ.

Substituting (7.3.3.1-2) into (7.3.3.1-1) finds for $\Delta \underline{R}_{SF_m}^{B}$:

For Non-Coning Angular Rate And Constant B Frame Specific Force:

$$\begin{aligned} \Delta \underline{R}_{SF_m}^{B(m-1)} = & \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \underline{a}_{SF}^{B}\, d\tau\, dt + \left(\underline{u}_\omega \times \underline{a}_{SF}^{B}\right) \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \sin\, \alpha(\tau)\, d\tau\, dt \\ & + \left[\underline{u}_\omega \times \left(\underline{u}_\omega \times \underline{a}_{SF}^{B}\right)\right] \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \left(1 - \cos\, \alpha(\tau)\right) d\tau\, dt \end{aligned} \tag{7.3.3.1-3}$$

For constant B Frame angular rate we see from (7.2.2.2-7) that:

For Constant B Frame Angular Rate:

$$\alpha(t) = \int_{t_{m-1}}^{t} \omega \, d\tau = \omega \, (t - t_{m-1}) \qquad \omega = \text{Constant.} \tag{7.3.3.1-4}$$

where

ω = Magnitude of $\underline{\omega}_{IB}^{B}$.

Applying (7.3.3.1-4) in (7.3.3.1-3) allows the integral terms containing $\alpha(\tau)$ to be evaluated as:

For Constant B Frame Angular Rate:

$$\int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \sin \, \alpha(\tau) \, d\tau \, dt = \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \sin \Big(\omega \, (\tau - t_{m-1})\Big) \, d\tau \, dt$$

$$= \int_{t_{m-1}}^{t_m} \frac{1}{\omega}\Big[1 - \cos \Big(\omega \, (t - t_{m-1})\Big)\Big] \, dt = \frac{1}{\omega}\left(T_m - \frac{1}{\omega} \sin \, (\omega T_m)\right) = \frac{1}{\omega^2}\Big(\alpha_m - \sin \, \alpha_m\Big)$$

$$\tag{7.3.3.1-5}$$

$$\int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \Big(1 - \cos \, \alpha(\tau)\Big) \, d\tau \, dt = \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \Big[1 - \cos \Big(\omega \, (\tau - t_{m-1})\Big)\Big] \, d\tau \, dt$$

$$= \int_{t_{m-1}}^{t_m} \left[(t - t_{m-1}) - \frac{1}{\omega} \sin \Big(\omega \, (t - t_{m-1})\Big)\right] dt = \frac{1}{2} T_m^2 - \frac{1}{\omega^2}\Big(1 - \cos \, (\omega T_m)\Big)$$

$$= \frac{1}{\omega^2}\left(\frac{1}{2} \alpha_m^2 - (1 - \cos \, \alpha_m)\right)$$

where

$\alpha_m = \alpha(t_m)$ from Equation (7.3.3.1-4).

With (7.3.3.1-4) we can also write:

$$\omega = \frac{\alpha_m}{T_m} \tag{7.3.3.1-6}$$

so that (7.3.3.1-5) becomes:

For Constant B Frame Angular Rate:

$$\int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \sin \alpha(\tau)\, d\tau\, dt = \frac{T_m^2}{\alpha_m}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right)$$

$$\int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \left(1 - \cos \alpha(\tau)\right) d\tau\, dt = T_m^2 \left(\frac{1}{2} - \frac{(1 - \cos \alpha_m)}{\alpha_m^2}\right) \tag{7.3.3.1-7}$$

Applying (7.3.3.1-7) in (7.3.3.1-3) then obtains:

For Constant B Frame Angular Rate And Specific Force:

$$\begin{aligned} \Delta \underline{R}_{SF_m}^{B(m-1)} = & \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \underline{a}_{SF}^{B}\, d\tau\, dt + \left(\underline{u}_\omega \times \underline{a}_{SF}^{B}\right) \frac{T_m^2}{\alpha_m}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) \\ & + \left[\underline{u}_\omega \times \left(\underline{u}_\omega \times \underline{a}_{SF}^{B}\right)\right] T_m^2 \left(\frac{1}{2} - \frac{(1 - \cos \alpha_m)}{\alpha_m^2}\right) \end{aligned} \tag{7.3.3.1-8}$$

Equation (7.3.3.1-8) can be further refined by substitution of $\underline{S}_{\upsilon_m}$ as defined in Equations (7.3.3-10) for the double integral, application of Equations (7.2.2.2-10), (7.2.2.2-14) and (7.3.3.1-6) for appropriate terms, and factorization:

For Constant B Frame Angular Rate And Specific Force:

$$\Delta \underline{R}_{SF_m}^{B(m-1)} = \underline{S}_{\upsilon_m} + \left[\frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) I + \frac{1}{\alpha_m^2}\left(\frac{1}{2} - \frac{(1 - \cos \alpha_m)}{\alpha_m^2}\right)\left(\underline{\alpha}_m \times\right)\right]\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) T_m \tag{7.3.3.1-9}$$

where

I = Identity matrix.

We also note for numerical computational purposes, that the Taylor series expansions for trigonometric functions in (7.3.3.1-9) are given by:

$$\frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) = \frac{1}{3\,!} - \frac{\alpha_m^2}{5\,!} + \frac{\alpha_m^4}{7\,!} - \cdots$$

$$\frac{1}{\alpha_m^2}\left(\frac{1}{2} - \frac{\left(1 - \cos \alpha_m\right)}{\alpha_m^2}\right) = \frac{1}{4\,!} - \frac{\alpha_m^2}{6\,!} + \frac{\alpha_m^4}{8\,!} - \cdots \tag{7.3.3.1-10}$$

The $\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) T_m$ term in (7.3.3.1-9) can be expressed in an alternative form through the following development. Using appropriate definitions from (7.3.3-10) we find that:

For Constant B Frame Angular Rate And Specific Force:

$$\underline{S}_{\upsilon_m} = \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \underline{a}_{SF}^{B} \, d\tau \, dt = \underline{a}_{SF}^{B} \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} d\tau \, dt = \frac{1}{2} \underline{a}_{SF}^{B} T_m^2 = \frac{1}{2} \underline{\upsilon}_m T_m$$

$$\underline{S}_{\alpha_m} = \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} \underline{\omega}_{IB}^{B} \, d\tau \, dt = \underline{\omega}_{IB}^{B} \int_{t_{m-1}}^{t_m} \int_{t_{m-1}}^{t} d\tau \, dt = \frac{1}{2} \underline{\omega}_{IB}^{B} T_m^2 = \frac{1}{2} \underline{\alpha}_m T_m \qquad (7.3.3.1\text{-}11)$$

From (7.3.3.1-11) we then can show that $\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) T_m$ under the (7.3.3.1-9) constant B Frame angular-rate/specific-force condition is equivalently:

For Constant B Frame Angular Rate And Specific Force:

$$\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) T_m = \underline{S}_{\alpha_m} \times \underline{\upsilon}_m + \underline{\alpha}_m \times \underline{S}_{\upsilon_m} \qquad (7.3.3.1\text{-}12)$$

We then substitute (7.3.3.1-12) for $\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) T_m$ in (7.3.3.1-9) to obtain:

For Constant B Frame Angular Rate And Specific Force:

$$\begin{aligned} \Delta\underline{R}_{SF_m}^{B(m-1)} = \underline{S}_{\upsilon_m} + \Bigg[& \frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) I \\ & + \frac{1}{\alpha_m^2}\left(\frac{1}{2} - \frac{(1 - \cos \alpha_m)}{\alpha_m^2}\right)\left(\underline{\alpha}_m \times\right) \Bigg] \left(\underline{S}_{\alpha_m} \times \underline{\upsilon}_m + \underline{\alpha}_m \times \underline{S}_{\upsilon_m}\right) \end{aligned} \qquad (7.3.3.1\text{-}13)$$

Equation (7.3.3.1-13) is now in a form for defining the exact position rotation compensation term by comparison with (7.3.3-9) for $\Delta\underline{R}_{SF_m}^{B(m-1)}$. Under the conditions of constant B Frame angular rate and specific force, the $\Delta\underline{R}_{Scrl_m}$ term in (7.3.3-9) is zero, and $\Delta\underline{R}_{SF_m}^{B(m-1)}$ with (7.3.3-11) reduces to:

For Constant B Frame Angular Rate And Specific Force:

$$\Delta\underline{R}_{SF_m}^{B(m-1)} = \underline{S}_{\upsilon_m} + \Delta\underline{R}_{Rot_m} \qquad \Delta\underline{R}_{Rot_m} = \frac{1}{6}\left(\underline{S}_{\alpha_m} \times \underline{\upsilon}_m + \underline{\alpha}_m \times \underline{S}_{\upsilon_m}\right) \qquad (7.3.3.1\text{-}14)$$

Applying Equations (7.3.3.1-10) shows that (7.3.3.1-13) to first order is given by:

For Constant B Frame Angular Rate And Specific Force:

$$\Delta \underline{R}_{SF_m}^{B(m-1)} = \underline{S}_{\upsilon m} + \frac{1}{6}\left(\underline{S}_{\alpha m} \times \underline{\upsilon}_m + \underline{\alpha}_m \times \underline{S}_{\upsilon m}\right) \qquad (7.3.3.1\text{-}15)$$

Finally, we compare (7.3.3.1-15) (i.e., the first order version of (7.3.3.1-13)) with (7.3.3.1-14) to deduce the sought after exact position rotation compensation algorithm from (7.3.3.1-13) with (7.3.3.1-10):

$$\Delta \underline{R}_{Rot_m} = \left[\frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) I + \frac{1}{\alpha_m^2}\left(\frac{1}{2} - \frac{(1 - \cos \alpha_m)}{\alpha_m^2}\right)\left(\underline{\alpha}_m \times\right)\right]\left(\underline{S}_{\alpha m} \times \underline{\upsilon}_m + \underline{\alpha}_m \times \underline{S}_{\upsilon m}\right)$$

$$\frac{1}{\alpha_m^2}\left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) = \frac{1}{3\,!} - \frac{\alpha_m^2}{5\,!} + \frac{\alpha_m^4}{7\,!} - \cdots \qquad (7.3.3.1\text{-}16)$$

$$\frac{1}{\alpha_m^2}\left(\frac{1}{2} - \frac{\left(1 - \cos \alpha_m\right)}{\alpha_m^2}\right) = \frac{1}{4\,!} - \frac{\alpha_m^2}{6\,!} + \frac{\alpha_m^4}{8\,!} - \cdots$$

Equations (7.3.3.1-16) can now be utilized in (7.3.3-9) for $\Delta \underline{R}_{Rot_m}$ in place of (7.3.3-11) to obtain the equivalent higher order equation for $\Delta \underline{R}_{SF_m}^{B(m-1)}$ that is exact under constant B Frame angular rate and specific force conditions.

Finally, it is to be noted that Chapter 19 (Section 19.1) describes a unified approach to strapdown algorithm design that uses a position translation vector (analogous to the rotation vector) for an exact calculation of $\Delta \underline{R}_{SF_m}^{B(m-1)}$. The position translation vector concept is formulated as an extension of Equation (7.3.3.1-9) to general motion (i.e., without requiring the restriction of constant B Frame angular-rate/specific-force vectors invoked in this section). The unified approach was developed by the author following the original publication of this book in 2000.

7.3.3.2 COMPUTER ALGORITHMS FOR SCROLLING AND OTHER INTEGRAL TERMS

The computer algorithms used to implement the integration operations in Equations (7.3.3-10) are executed at a high computation cycle rate within the position update cycle. The $\underline{\alpha}_m$ and $\underline{\upsilon}_m$ integral terms in (7.3.3-10) are provided by Equations (7.2.2.2.2-4). The remaining integral terms in (7.3.3-10) can be rewritten to reflect the high speed computing cycle as follows:

$$\underline{\alpha}(\tau) = \underline{\alpha}_{l-1} + \Delta\underline{\alpha}(\tau)$$

$$\Delta\underline{\alpha}(\tau) = \int_{t_{l-1}}^{\tau} \underline{\omega}_{IB}^{B}\, dt \qquad \Delta\underline{\alpha}_{l} = \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}^{B}\, dt$$

$$\underline{\alpha}_{l} = \underline{\alpha}_{l-1} + \Delta\underline{\alpha}_{l} \qquad \underline{\alpha}_{l} = 0 \ \text{At} \ \tau = t_{m-1}$$

$$\underline{\upsilon}(\tau) = \underline{\upsilon}_{l-1} + \Delta\underline{\upsilon}(\tau)$$

$$\Delta\underline{\upsilon}(\tau) = \int_{t_{l-1}}^{\tau} \underline{a}_{SF}^{B}\, dt \qquad \Delta\underline{\upsilon}_{l} = \int_{t_{l-1}}^{t_l} \underline{a}_{SF}^{B}\, dt$$

$$\underline{\upsilon}_{l} = \underline{\upsilon}_{l-1} + \Delta\underline{\upsilon}_{l} \qquad \underline{\upsilon}_{l} = 0 \ \text{At} \ \tau = t_{m-1}$$

$$\underline{S}_{\alpha}(t) = \underline{S}_{\alpha_{l-1}} + \Delta\underline{S}_{\alpha}(t) \tag{7.3.3.2-1}$$

$$\Delta\underline{S}_{\alpha}(t) = \int_{t_{l-1}}^{t} \underline{\alpha}(\tau)\, d\tau \qquad \Delta\underline{S}_{\alpha_l} = \Delta\underline{S}_{\alpha}(t_l)$$

$$\underline{S}_{\alpha_l} = \underline{S}_{\alpha_{l-1}} + \Delta\underline{S}_{\alpha_l} \qquad \underline{S}_{\alpha_m} = \underline{S}_{\alpha_l}(t_l = t_m) \qquad \underline{S}_{\alpha_l} = 0 \ \text{At} \ t = t_{m-1}$$

$$\underline{S}_{\upsilon}(t) = \underline{S}_{\upsilon_{l-1}} + \Delta\underline{S}_{\upsilon}(t)$$

$$\Delta\underline{S}_{\upsilon}(t) = \int_{t_{l-1}}^{t} \underline{\upsilon}(\tau)\, d\tau \qquad \Delta\underline{S}_{\upsilon_l} = \Delta\underline{S}_{\upsilon}(t_l)$$

$$\underline{S}_{\upsilon_l} = \underline{S}_{\upsilon_{l-1}} + \Delta\underline{S}_{\upsilon_l} \qquad \underline{S}_{\upsilon_m} = \underline{S}_{\upsilon_l}(t_l = t_m) \qquad \underline{S}_{\upsilon_l} = 0 \ \text{At} \ t = t_{m-1}$$

$$\Delta\underline{R}_{Scrl_l} = \Delta\underline{R}_{Scrl_{l-1}} + \delta\underline{R}_{Scrl_l} \qquad \Delta\underline{R}_{Scrl_m} = \Delta\underline{R}_{Scrl_l}(t_l = t_m) \qquad \Delta\underline{R}_{Scrl_l} = 0 \ \text{At} \ t = t_{m-1}$$

$$\delta\underline{R}_{Scrl_l} = \int_{t_{l-1}}^{t_l} \frac{1}{6}\left[6\, \Delta\underline{v}_{Scul}(t) - \underline{S}_{\alpha}(t) \times \underline{a}_{SF}^{B} + \underline{S}_{\upsilon}(t) \times \underline{\omega}_{IB}^{B} + \underline{\alpha}(t) \times \underline{\upsilon}(t)\right] dt$$

$$\Delta\underline{v}_{Scul}(t) = \Delta\underline{v}_{Scul_{l-1}} + \delta\underline{v}_{Scul}(t) \qquad \Delta\underline{v}_{Scul_l} = \Delta\underline{v}_{Scul}(t_l) \qquad \Delta\underline{v}_{Scul_l} = 0 \ \text{At} \ t = t_{m-1}$$

$$\delta\underline{v}_{Scul}(t) = \int_{t_{l-1}}^{t} \frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) d\tau$$

where

l = High speed computer cycle index.

t_{l-1}, t_l = Time at computer cycles l-1 and l.

As in Section 7.2.2.2.2 for the velocity sculling algorithm and other integral terms, algorithms can designed for the integral terms in (7.3.3.2-1) to be analytically exact under assumed forms of the angular rate and specific force acceleration profile within the l cycle. Coefficients for the angular-rate/specific-force profiles are then determined from sequential integrated angular-rate/specific-force increments taken at the l cycle rate (or, alternatively, at a higher speed sensor sampling rate within the l cycle). For this section, we provide an example based on general linearly changing angular-rate/specific-force over the t_{l-1} to t_l time interval, the coefficients of which are calculated from current and past l cycle sensor samples (as in Section 7.2.2.2.2). Thus, we approximate:

$$\underline{\omega}_{IB}^{B} \approx \underline{A} + \underline{B}\left(t - t_{l-1}\right) \qquad \underline{a}_{SF}^{B} \approx \underline{C} + \underline{D}\left(t - t_{l-1}\right) \tag{7.3.3.2-2}$$

where

$\underline{A}, \underline{B}, \underline{C}, \underline{D}$ = Constant vectors.

Substituting (7.3.3.2-2) into (7.3.3.2-1) yields the following for particular terms:

$$\begin{aligned}
\Delta\underline{\alpha}(t) &= \left(\underline{A} + \frac{1}{2}\,\underline{B}\,(t - t_{l-1})\right)(t - t_{l-1}) \\
\Delta\underline{\upsilon}(t) &= \left(\underline{C} + \frac{1}{2}\,\underline{D}\,(t - t_{l-1})\right)(t - t_{l-1}) \\
\underline{\alpha}(t) &= \underline{\alpha}_{l-1} + \underline{A}\,(t - t_{l-1}) + \frac{1}{2}\,\underline{B}\,(t - t_{l-1})^2 \\
\underline{\upsilon}(t) &= \underline{\upsilon}_{l-1} + \underline{C}\,(t - t_{l-1}) + \frac{1}{2}\,\underline{D}\,(t - t_{l-1})^2 \\
\Delta\underline{S}_{\alpha}(t) &= \underline{\alpha}_{l-1}\,(t - t_{l-1}) + \frac{1}{2}\left(\underline{A} + \frac{1}{3}\,\underline{B}\,(t - t_{l-1})\right)(t - t_{l-1})^2 \\
\Delta\underline{S}_{\upsilon}(t) &= \underline{\upsilon}_{l-1}\,(t - t_{l-1}) + \frac{1}{2}\left(\underline{C} + \frac{1}{3}\,\underline{D}\,(t - t_{l-1})\right)(t - t_{l-1})^2 \\
\underline{S}_{\alpha}(t) &= \underline{S}_{\alpha_{l-1}} + \underline{\alpha}_{l-1}\,(t - t_{l-1}) + \frac{1}{2}\,\underline{A}\,(t - t_{l-1})^2 + \frac{1}{6}\,\underline{B}\,(t - t_{l-1})^3 \\
\underline{S}_{\upsilon}(t) &= \underline{S}_{\upsilon_{l-1}} + \underline{\upsilon}_{l-1}\,(t - t_{l-1}) + \frac{1}{2}\,\underline{C}\,(t - t_{l-1})^2 + \frac{1}{6}\,\underline{D}\,(t - t_{l-1})^3
\end{aligned} \tag{7.3.3.2-3}$$

With (7.3.3.2-2) and (7.3.3.2-3), the $\delta\underline{v}_{Scul}(t)$ integrand in (7.3.3.2-1) becomes:

$$
\begin{aligned}
&\frac{1}{2}\left(\underline{\alpha}(\tau) \times \underline{a}_{SF}^{B} + \underline{\upsilon}(\tau) \times \underline{\omega}_{IB}^{B}\right) \\
&\quad = \frac{1}{2}\left(\underline{\alpha}_{l-1} + \underline{A}\,(\tau\text{-}t_{l-1}) + \frac{1}{2}\,\underline{B}\,(\tau\text{-}t_{l-1})^{2}\right) \times \left(\underline{C} + \underline{D}\,(\tau\text{-}t_{l-1})\right) \\
&\qquad + \frac{1}{2}\left(\underline{\upsilon}_{l-1} + \underline{C}\,(\tau\text{-}t_{l-1}) + \frac{1}{2}\,\underline{D}\,(\tau\text{-}t_{l-1})^{2}\right) \times \left(\underline{A} + \underline{B}\,(\tau\text{-}t_{l-1})\right) \\
&= \frac{1}{2}\,\underline{\alpha}_{l-1} \times \left(\underline{C} + \underline{D}\,(\tau\text{-}t_{l-1})\right) + \frac{1}{2}\,\underline{\upsilon}_{l-1} \times \left(\underline{A} + \underline{B}\,(\tau\text{-}t_{l-1})\right) + \frac{1}{4}\left(\underline{A} \times \underline{D} + \underline{C} \times \underline{B}\right)(\tau\text{-}t_{l-1})^{2}
\end{aligned}
\tag{7.3.3.2-4}
$$

Taking the integral of (7.3.3.2-4) from t_{l-1} to t obtains for $\Delta \underline{v}_{Scul}(t)$ in (7.3.3.2-1):

$$
\begin{aligned}
\Delta \underline{v}_{Scul}(t) \;=\; & \Delta \underline{v}_{Scul_{l-1}} + \frac{1}{2}\,\underline{\alpha}_{l-1} \times \left(\underline{C} + \frac{1}{2}\,\underline{D}\,(t\text{-}t_{l-1})\right)(t\text{-}t_{l-1}) \\
& + \frac{1}{2}\,\underline{\upsilon}_{l-1} \times \left(\underline{A} + \frac{1}{2}\,\underline{B}\,(t\text{-}t_{l-1})\right)(t\text{-}t_{l-1}) + \frac{1}{12}\left(\underline{A} \times \underline{D} + \underline{C} \times \underline{B}\right)(t\text{-}t_{l-1})^{3}
\end{aligned}
\tag{7.3.3.2-5}
$$

The integral of (7.3.3.2-5) for $\delta \underline{R}_{Scrl_l}$ in Equation (7.3.3.2-1) is then given by:

$$
\begin{aligned}
\int_{t_{l-1}}^{t_l} \Delta \underline{v}_{Scul}(t)\,dt \;=\; & \Delta \underline{v}_{Scul_{l-1}} T_l + \frac{1}{2}\,\underline{\alpha}_{l-1} \times \left(\frac{1}{2}\,\underline{C} + \frac{1}{6}\,\underline{D}\,T_l\right) T_l^{2} \\
& + \frac{1}{2}\,\underline{\upsilon}_{l-1} \times \left(\frac{1}{2}\,\underline{A} + \frac{1}{6}\,\underline{B}\,T_l\right) T_l^{2} + \frac{1}{48}\left(\underline{A} \times \underline{D} + \underline{C} \times \underline{B}\right) T_l^{4} \\
=\; & \Delta \underline{v}_{Scul_{l-1}} T_l + \frac{1}{2}\left(\underline{\alpha}_{l-1} - \frac{1}{12}\,\underline{B}\,T_l^{2}\right) \times \left(\frac{1}{2}\,\underline{C} + \frac{1}{6}\,\underline{D}\,T_l\right) T_l^{2} \\
& + \frac{1}{2}\left(\underline{\upsilon}_{l-1} - \frac{1}{12}\,\underline{D}\,T_l^{2}\right) \times \left(\frac{1}{2}\,\underline{A} + \frac{1}{6}\,\underline{B}\,T_l\right) T_l^{2}
\end{aligned}
\tag{7.3.3.2-6}
$$

where

T_l = High speed computer l cycle update time interval t_l - t_{l-1}.

With Equation (7.3.3.2-3) for particular terms, Equation (7.3.3.2-6) becomes:

$$
\begin{aligned}
\int_{t_{l-1}}^{t_l} \Delta \underline{v}_{Scul}(t)\,dt \;=\; & \Delta \underline{v}_{Scul_{l-1}} T_l + \frac{1}{2}\left(\underline{\alpha}_{l-1} - \frac{1}{12}\,\underline{B}\,T_l^{2}\right) \times \left(\Delta \underline{S}_{\upsilon_l} - \underline{\upsilon}_{l-1} T_l\right) \\
& + \frac{1}{2}\left(\underline{\upsilon}_{l-1} - \frac{1}{12}\,\underline{D}\,T_l^{2}\right) \times \left(\Delta \underline{S}_{\alpha_l} - \underline{\alpha}_{l-1} T_l\right)
\end{aligned}
\tag{7.3.3.2-7}
$$

Following a similar development using Equations (7.3.3.2-2) - (7.3.3.2-3) for the $\delta \underline{R}_{Scrl_l}$ integrand terms in (7.3.3.2-1) (exclusive of the $\Delta \underline{v}_{Scul}$ element) yields:

$$
\begin{aligned}
&- \underline{S}_{\alpha}(t) \times \underline{a}_{SF}^{B} + \underline{S}_{\upsilon}(t) \times \underline{\omega}_{IB}^{B} + \underline{\alpha}(t) \times \underline{\upsilon}(t) \\
&\quad = - \left(\underline{S}_{\alpha_{l-1}} + \underline{\alpha}_{l-1} \, (t - t_{l-1}) + \frac{1}{2} \, \underline{A} \, (t - t_{l-1})^2 + \frac{1}{6} \, \underline{B} \, (t - t_{l-1})^3 \right) \times \left(\underline{C} + \underline{D} \, (t - t_{l-1}) \right) \\
&\qquad + \left(\underline{S}_{\upsilon_{l-1}} + \underline{\upsilon}_{l-1} \, (t - t_{l-1}) + \frac{1}{2} \, \underline{C} \, (t - t_{l-1})^2 + \frac{1}{6} \, \underline{D} \, (t - t_{l-1})^3 \right) \times \left(\underline{A} + \underline{B} \, (t - t_{l-1}) \right) \\
&\qquad + \left(\underline{\alpha}_{l-1} + \underline{A} \, (t - t_{l-1}) + \frac{1}{2} \, \underline{B} \, (t - t_{l-1})^2 \right) \times \left(\underline{\upsilon}_{l-1} + \underline{C} \, (t - t_{l-1}) + \frac{1}{2} \, \underline{D} \, (t - t_{l-1})^2 \right) \qquad (7.3.3.2\text{-}8) \\
&\quad = - \underline{S}_{\alpha_{l-1}} \times \underline{a}_{SF}^{B} + \underline{S}_{\upsilon_{l-1}} \times \underline{\omega}_{IB}^{B} + \underline{\alpha}_{l-1} \times \underline{\upsilon}_{l-1} \\
&\qquad + \frac{1}{2} \left(\underline{\upsilon}_{l-1} \times \underline{B} - \underline{\alpha}_{l-1} \times \underline{D} \right) (t - t_{l-1})^2 \\
&\qquad + \frac{1}{6} \left(\underline{D} \times \underline{A} + \underline{C} \times \underline{B} \right) (t - t_{l-1})^3 + \frac{1}{12} \left(\underline{D} \times \underline{B} \right) (t - t_{l-1})^4
\end{aligned}
$$

Integrating (7.3.3.2-8) and substitution with (7.3.3.2-7) into (7.3.3.2-1) then obtains for $\delta \underline{R}_{Scrl_l}$:

$$
\delta \underline{R}_{Scrl_l} = \delta \underline{R}_{ScrlA_l} + \delta \underline{R}_{ScrlB_l}
$$

$$
\begin{aligned}
\delta \underline{R}_{ScrlA_l} &= \Delta \underline{v}_{Scul_{l-1}} \, T_l + \frac{1}{2} \left(\underline{\alpha}_{l-1} - \frac{1}{12} \, \underline{B} \, T_l^2 \right) \times \left(\Delta \underline{S}_{\upsilon_l} - \underline{\upsilon}_{l-1} \, T_l \right) \\
&\quad + \frac{1}{2} \left(\underline{\upsilon}_{l-1} - \frac{1}{12} \, \underline{D} \, T_l^2 \right) \times \left(\Delta \underline{S}_{\alpha_l} - \underline{\alpha}_{l-1} \, T_l \right) \qquad (7.3.3.2\text{-}9) \\
\delta \underline{R}_{ScrlB_l} &= \frac{1}{6} \left(- \underline{S}_{\alpha_{l-1}} \times \Delta \underline{\upsilon}_l + \underline{S}_{\upsilon_{l-1}} \times \Delta \underline{\alpha}_l \right) + \frac{1}{6} \left(\underline{\alpha}_{l-1} \times \underline{\upsilon}_{l-1} \right) T_l \\
&\quad + \frac{1}{36} \left(\underline{\upsilon}_{l-1} \times \underline{B} - \underline{\alpha}_{l-1} \times \underline{D} \right) T_l^3 \\
&\quad + \frac{1}{144} \left(\underline{D} \times \underline{A} + \underline{C} \times \underline{B} \right) T_l^4 + \frac{1}{360} \left(\underline{D} \times \underline{B} \right) T_l^5
\end{aligned}
$$

where

$\delta \underline{R}_{ScrlA_l}$ = Portion of $\delta \underline{R}_{Scrl_l}$ produced by the $\Delta \underline{v}_{Scul}$ sculling term.

$\delta \underline{R}_{ScrlB_l}$ = Portion of $\delta \underline{R}_{Scrl_l}$ produced by all but the $\Delta \underline{v}_{Scul}$ sculling term.

It finally remains to define the $\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$ terms in (7.3.3.2-9) in terms of strapdown sensor input parameters. As in Equations (7.2.2.2.2-10), assuming approximately linearly ramping angular rates and accelerations, these terms can be defined in terms of current and past values of the $\Delta \underline{\alpha}$, $\Delta \underline{\upsilon}$ integrated strapdown angular-rate/specific-force increments:

$$\underline{A} = \frac{1}{2\,T_l}\left(\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) \qquad \underline{B} = \frac{1}{T_l^2}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)$$
$$\underline{C} = \frac{1}{2\,T_l}\left(\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right) \qquad \underline{D} = \frac{1}{T_l^2}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) \tag{7.3.3.2-10}$$

Substitution of (7.3.3.2-10) into the last two terms of the $\delta\underline{R}_{\mathrm{ScrlB}_l}$ expression in Equation (7.3.3.2-9) yields:

$$\frac{1}{144}\left(\underline{D}\times\underline{A} + \underline{C}\times\underline{B}\right)T_l^4 + \frac{1}{360}\left(\underline{D}\times\underline{B}\right)T_l^5$$

$$= \frac{T_l}{288}\left[\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\times\left(\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) + \left(\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right)\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right]$$
$$+ \frac{T_l}{360}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right) \tag{7.3.3.2-11}$$

$$= \frac{T_l}{288}\left\{\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\times\left[2\,\Delta\underline{\alpha}_l - \left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] + \left[2\,\Delta\underline{\upsilon}_l - \left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right]\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right\}$$
$$+ \frac{T_l}{360}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)$$

$$= \frac{T_l}{144}\left[\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\times\Delta\underline{\alpha}_l + \Delta\underline{\upsilon}_l\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] - T_l\left(\frac{1}{144} - \frac{1}{360}\right)\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)$$

$$= \frac{T_l}{144}\left[\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\times\Delta\underline{\alpha}_l + \Delta\underline{\upsilon}_l\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] - \frac{T_l}{240}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)$$

With (7.3.3.2-10), the second and third terms in the Equations (7.3.3.2-9) $\delta\underline{R}_{\mathrm{ScrlB}_l}$ expression are:

$$\frac{1}{6}\left(\underline{\alpha}_{l-1}\times\underline{\upsilon}_{l-1}\right)T_l + \frac{1}{36}\left(\underline{\upsilon}_{l-1}\times\underline{B} - \underline{\alpha}_{l-1}\times\underline{D}\right)T_l^3$$
$$= \frac{T_l}{6}\left(\underline{\alpha}_{l-1}\times\underline{\upsilon}_{l-1}\right) + \frac{T_l}{36}\left[\underline{\upsilon}_{l-1}\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right) - \underline{\alpha}_{l-1}\times\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right]$$
$$= \frac{T_l}{6}\left[\underline{\alpha}_{l-1} - \frac{1}{6}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right]\times\left[\underline{\upsilon}_{l-1} - \frac{1}{6}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] \tag{7.3.3.2-12}$$
$$+ \frac{T_l}{216}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\times\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)$$

Equations (7.3.3.2-11) and (7.3.3.2-12) in combination become:

$$\frac{1}{6}\left(\underline{\alpha}_{l-1} \times \underline{\upsilon}_{l-1}\right) T_l + \frac{1}{36}\left(\underline{\upsilon}_{l-1} \times \underline{B} - \underline{\alpha}_{l-1} \times \underline{D}\right) T_l^3 + \frac{1}{144}\left(\underline{D} \times \underline{A} + \underline{C} \times \underline{B}\right) T_l^4 + \frac{1}{360}\left(\underline{D} \times \underline{B}\right) T_l^5$$

$$\begin{aligned} &= \frac{T_l}{144}\left[\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) \times \Delta\underline{\alpha}_l + \Delta\underline{\upsilon}_l \times \left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \\ &\quad + \frac{T_l}{6}\left[\underline{\alpha}_{l-1} - \frac{1}{6}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \times \left[\underline{\upsilon}_{l-1} - \frac{1}{6}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] \\ &\quad + \frac{T_l}{2160}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) \times \left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right) \end{aligned} \tag{7.3.3.2-13}$$

Finally, we substitute (7.3.3.2-13) into (7.3.3.2-9) to obtain the desired form for the $\delta\underline{R}_{ScrlB_l}$ expression. With (7.3.3.2-10) substitution into the $\delta\underline{R}_{ScrlA_l}$ expression, the final result is:

$$\begin{aligned} \delta\underline{R}_{Scrl_l} &= \delta\underline{R}_{ScrlA_l} + \delta\underline{R}_{ScrlB_l} \\ \delta\underline{R}_{ScrlA_l} &= \Delta\underline{v}_{Scul_{l-1}}\, T_l + \frac{1}{2}\left[\underline{\alpha}_{l-1} - \frac{1}{12}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \times \left(\Delta\underline{S}_{\upsilon_l} - \underline{\upsilon}_{l-1}\, T_l\right) \\ &\quad + \frac{1}{2}\left[\underline{\upsilon}_{l-1} - \frac{1}{12}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] \times \left(\Delta\underline{S}_{\alpha_l} - \underline{\alpha}_{l-1}\, T_l\right) \\ \delta\underline{R}_{ScrlB_l} &= \frac{1}{6}\left[\underline{S}_{\upsilon_{l-1}} + \frac{T_l}{24}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] \times \Delta\underline{\alpha}_l \\ &\quad - \frac{1}{6}\left[\underline{S}_{\alpha_{l-1}} + \frac{T_l}{24}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \times \Delta\underline{\upsilon}_l \\ &\quad + \frac{T_l}{6}\left[\underline{\alpha}_{l-1} - \frac{1}{6}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \times \left[\underline{\upsilon}_{l-1} - \frac{1}{6}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] \\ &\quad - \frac{T_l}{2160}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right) \times \left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) \end{aligned} \tag{7.3.3.2-14}$$

Algorithms for evaluating $\Delta\underline{S}_{\alpha_l}$ and $\Delta\underline{S}_{\upsilon_l}$ in Equations (7.3.3.2-1) are derived similarly by substitution of (7.3.3.2-10) into (7.3.3.2-3). The development for $\Delta\underline{S}_{\alpha_l}$ is as follows:

$$\begin{aligned} \Delta\underline{S}_{\alpha_l} &= \underline{\alpha}_{l-1} T_l + \frac{1}{2}\left[\frac{1}{2}\left(\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) + \frac{1}{3}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] T_l \\ &= \underline{\alpha}_{l-1} T_l + \frac{1}{12}\left(5\ \Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) T_l \end{aligned} \tag{7.3.3.2-15}$$

The result for $\Delta\underline{S}_{\alpha_l}$ and $\Delta\underline{S}_{\upsilon_l}$ then is:

$$\begin{aligned} \Delta\underline{S}_{\alpha_l} &= \underline{\alpha}_{l-1} T_l + \frac{T_l}{12}\left(5\ \Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right) \\ \Delta\underline{S}_{\upsilon_l} &= \underline{\upsilon}_{l-1} T_l + \frac{T_l}{12}\left(5\ \Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right) \end{aligned} \tag{7.3.3.2-16}$$

In summary, the algorithm used for calculating $\Delta\underline{R}_{Scrl_m}$ in Equation (7.3.3-9) is given by Equations (7.3.3.2-1) with (7.3.3.2-14) and (7.3.3.2-16). The $\Delta\underline{v}_{Scul_{l-1}}$ term in (7.3.3.2-14) is provided from the sculling algorithm in Equations (7.2.2.2.2-15). In combination, the overall composite result for $\Delta\underline{R}_{Scrl_m}$ as well as the $\underline{S}_{\alpha_m}$, $\underline{\alpha}_m$, $\underline{S}_{\upsilon_m}$, $\underline{\upsilon}_m$ terms for Equations (7.3.3-9) and (7.3.3-11) or (7.3.3.1-16) is as follows:

$$
\begin{aligned}
\Delta\underline{\alpha}_l,\ \underline{\alpha}_l &= \begin{array}{l}\text{Integrated Angular Rate Sensor Outputs}\\ \text{From Algorithm Equations (7.1.1.1.1-17)}\end{array}\\
\Delta\underline{\upsilon}_l,\ \underline{\upsilon}_l &= \begin{array}{l}\text{Integrated Accelerometer Outputs}\\ \text{From Algorithm Equations (7.2.2.2.2-14)}\end{array}
\end{aligned}
\tag{7.3.3.2-17}
$$

$$
\begin{aligned}
&\Delta\underline{S}_{\alpha_l} = \underline{\alpha}_{l-1} T_l + \frac{T_l}{12}\left(5\,\Delta\underline{\alpha}_l + \Delta\underline{\alpha}_{l-1}\right)\\
&\underline{S}_{\alpha_l} = \underline{S}_{\alpha_{l-1}} + \Delta\underline{S}_{\alpha_l} \qquad \underline{S}_{\alpha_m} = \underline{S}_{\alpha_l}(t_l = t_m) \qquad \underline{S}_{\alpha_l} = 0 \ \text{ At } \ t = t_{m-1}
\end{aligned}
\tag{7.3.3.2-18}
$$

$$
\begin{aligned}
&\Delta\underline{S}_{\upsilon_l} = \underline{\upsilon}_{l-1} T_l + \frac{T_l}{12}\left(5\,\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right)\\
&\underline{S}_{\upsilon_l} = \underline{S}_{\upsilon_{l-1}} + \Delta\underline{S}_{\upsilon_l} \qquad \underline{S}_{\upsilon_m} = \underline{S}_{\upsilon_l}(t_l = t_m) \qquad \underline{S}_{\upsilon_l} = 0 \ \text{ At } \ t = t_{m-1}
\end{aligned}
\tag{7.3.3.2-19}
$$

$$
\begin{aligned}
\Delta\underline{R}_{Scrl_l} &= \Delta\underline{R}_{Scrl_{l-1}} + \delta\underline{R}_{ScrlA_l} + \delta\underline{R}_{ScrlB_l}\\
\delta\underline{R}_{ScrlA_l} &= \Delta\underline{v}_{Scul_{l-1}}\, T_l + \frac{1}{2}\left[\underline{\alpha}_{l-1} - \frac{1}{12}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \times \left(\Delta\underline{S}_{\upsilon_l} - \underline{\upsilon}_{l-1}\, T_l\right)\\
&\quad + \frac{1}{2}\left[\underline{\upsilon}_{l-1} - \frac{1}{12}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] \times \left(\Delta\underline{S}_{\alpha_l} - \underline{\alpha}_{l-1}\, T_l\right)\\
\delta\underline{R}_{ScrlB_l} &= \frac{1}{6}\left[\underline{S}_{\upsilon_{l-1}} + \frac{T_l}{24}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] \times \Delta\underline{\alpha}_l\\
&\quad - \frac{1}{6}\left[\underline{S}_{\alpha_{l-1}} + \frac{T_l}{24}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \times \Delta\underline{\upsilon}_l\\
&\quad + \frac{T_l}{6}\left[\underline{\alpha}_{l-1} - \frac{1}{6}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)\right] \times \left[\underline{\upsilon}_{l-1} - \frac{1}{6}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right]\\
&\quad - \frac{T_l}{2160}\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right) \times \left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\\
\Delta\underline{v}_{Scul_{l-1}} &= \text{From Sculling Algorithm Equation (7.2.2.2.2-15)}\\
\Delta\underline{R}_{Scrl_m} &= \Delta\underline{R}_{Scrl_l}(t_l = t_m) \qquad \Delta\underline{R}_{Scrl_l} = 0 \ \text{ At } \ t = t_{m-1}
\end{aligned}
\tag{7.3.3.2-20}
$$

Equations (7.3.3.2-20) can be classified as a second order algorithm for $\delta\underline{R}_{Scrl_l}$ because it includes current and past cycle $\Delta\underline{\alpha}_l$, $\Delta\underline{\upsilon}_l$ products. If the angular-rate/specific-force profile was approximated as constant over two successive l cycles, the $\left(\Delta\underline{\alpha}_l - \Delta\underline{\alpha}_{l-1}\right)$, $\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)$ terms in (7.3.3.2-20) would vanish, resulting in a first order $\delta\underline{R}_{Scrl_l}$ algorithm. Under conditions when the angular rate and specific force acceleration can be approximated as constant (i.e., slowly varying) over an m cycle, $\Delta\underline{R}_{Scrl_l}$ in (7.3.3.2-20) is approximately zero and the $\Delta\underline{R}_{Scrl_l}$, $\delta\underline{R}_{ScrlA_l}$, $\delta\underline{R}_{ScrlB_l}$ calculations in (7.3.3.2-20) can be deleted. Alternatively (and more accurately), for slowly varying angular rates and accelerations, one l cycle of Equations (7.3.3.2-20) can be executed each m cycle, while noting from the initial condition definitions that $\underline{\alpha}_{l-1}$, $\underline{\upsilon}_{l-1}$, $\underline{S}_{\alpha_{l-1}}$ and $\underline{S}_{\upsilon_{l-1}}$ are zero. As noted in Sections 7.1.1.1.1 and 7.2.2.2.2, setting the l and m rates equal can also be achieved by increasing the m rate to match the l rate. The result would be a single high speed higher order algorithm with a simpler software architecture than the two-speed approach, but requiring more throughput. Continuing advances in the speed of modern day computers may make this the preferred approach for the future.

It is also to be noted that Chapter 19 (Section 19.1) describes a unified approach to strapdown algorithm design that uses a position translation vector (analogous to the rotation vector) for an exact calculation of $\Delta\underline{R}_{SF_m}^{B_{m-1}}$. The position translation vector concept is formulated as an extension of Equation (7.3.3.1-9) to general motion (i.e., without requiring the restriction of constant B Frame angular-rate/specific-force vectors invoked in this section). General motion is accounted for in the unified approach using a high speed scrolling calculation as part of the position translation vector computation. The unified scrolling equation is similar to but not identical to the scrolling equations of this section. Both scrolling concepts are identically zero under constant B Frame angular-rate/specific-force. The unified approach was developed by the author following the original publication of this book in 2000.

7.4 ALGORITHM AND EXECUTION RATE SELECTION

Faced with the multitude of potential strapdown inertial navigation algorithms to choose from, the software designer must ultimately choose one set for the application at hand. The algorithms presented in Sections 7.1 - 7.3 are but one version of many similar algorithms developed over the years by several authors. The process of selecting the algorithms and their execution rates for a particular application should consider the allowable algorithm error under anticipated angular-rate/specific-force vibration, the capability of the projected target navigation computer for the required algorithm execution rate, and the complexity of the design procedure for software validation/documentation with the selected algorithms.

For the two-speed attitude/velocity/position algorithm updating approach described in Sections 7.1 - 7.3, the repetition rate for the moderate speed (m cycle) algorithms would be typically selected based on maximum angular-rate/specific-force considerations, to minimize power series truncation error in the moderate and high speed algorithms. The repetition rate for the high speed (l cycle) algorithms would be typically selected based on the anticipated strapdown inertial sensor assembly vibration environment, to accurately account for vibration induced coning/sculling/scrolling effects.

Evaluation of candidate algorithm error characteristics is generally performed using computerized time domain simulators that exercise the algorithms in particular groupings at their selected repetition rates. The simulators generate simulated strapdown inertial sensor angular-rate/specific-force profiles for algorithm test input, together with known navigation parameter solutions for algorithm output comparison (e.g., Section 11.2.). For the attitude/velocity algorithms discussed in Sections 7.1 and 7.2, simplified analytical error models can also be used to predict high speed coning/sculling algorithm error under specified coning/sculling rates/amplitudes as a function of algorithm repetition rate (See Chapter 10). The coning/sculling rates/amplitudes must be derived either from empirical data, or more commonly, from analytical models of the sensor assembly mount imbalance and its response to external input vibration at particular frequencies (as in Chapter 10). Frequency domain simulators (e.g., Section 10.6) can be used to evaluate high speed coning/sculling algorithm error under specified input vibration power spectral density profiles and sensor assembly mount imbalance as a function of algorithm repetition rate. For example, the coning/sculling algorithms described by Equations (7.1.1.1.1-18) and (7.2.2.2.2-15) can be shown by Equations (10.6.1-25) to have errors of 0.00037 deg/hr ($\mathcal{E}\left(\delta\dot{\beta}_{\text{Algo-m}_z}\right)$ for the coning algorithm) and 0.044 micro-g's ($\mathcal{E}\left(\delta\Delta\dot{v}_{\text{SculAlgo-m}_z}\right)$ for the sculling algorithm) when operated at a 2 KHz repetition rate (T_l^{-1}) under exposure to 7.6 g's root-mean-square $\sqrt{\mathcal{E}\left(a_{\text{Vib}}^2(t)\right)}$ wide band random linear input vibration (flat $G_{a\text{Vib}}(\omega)$ at 0.04 g^2/Hz density from 20 - 1000 Hz, then decreasing logarithmically to 0.01 g^2/Hz at 2000 Hz). The linear vibration generates a multi-axis 3.6 g / 0.38 milli-radian root-mean-square linear-acceleration/angular-displacement oscillation ($\sqrt{\mathcal{E}\left(\overline{a_{SF}(t)^2}\right)}$ and $\sqrt{\mathcal{E}\left(\overline{\theta(t)^2}\right)}$) of the sensor assembly with corresponding coning/sculling rates of 9.9 deg/hr ($\mathcal{E}\left(\dot{\Phi}_{\text{Con}_z}\right)$) and 1.3 milli-g's ($\mathcal{E}\left(\dot{v}_{\text{SF/Scul}_z}\right)$) due to the following typical sensor assembly mount characteristics input to Equations (10.6.1-8) as simulator input parameters: 50 Hz linear vibration mode undamped natural frequency ($\omega_x / 2\,\pi$), 0.125 linear vibration mode damping ratio (ζ_x), 71 Hz rotary vibration mode undamped natural frequency ($\omega_\theta / 2\,\pi$), 0.18 rotary vibration mode damping ratio (ζ_θ), 5% sensor assembly mount/isolator spring/damping imbalance ($\varepsilon_k \times 100$ and $\varepsilon_c \times 100$), 1.4% sensor assembly center of mass offset from the isolator/mount center of force ($\varepsilon_l \times 100$ percent of distance between isolators) and 7 inches distance between sensor assembly isolators

($L \times 12$). Similar analysis can also be used to evaluate the "folding effect" error in the Section 7.3 position integration algorithms (see Section 10.1.3.2.3) using the $\mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right)$ expression in Equations (10.6.1-25).

The capabilities of modern day computer and INS software technology make it reasonable to specify that the attitude algorithm error be no greater than 5% of the equivalent error produced by the INS inertial sensors (whose cost increases dramatically with accuracy demands). For an INS with sensor bias accuracy requirements of 0.007 deg/hr for the angular rate sensors and 40 micro-g for the accelerometers (typical for an aircraft INS having 1 nautical mile per hour 50 percentile radial position error rate and 2 - 3 fps 1 sigma velocity accuracy), the above 0.00037 deg/hr coning algorithm error satisfies the 5 % allowance, while the above 0.044 micro-g sculling algorithm error is almost two orders of magnitude within the 5% allowance. For this situation, a 1 KHz sculling algorithm rate would probably be more appropriate, however, 2 KHz might still be utilized for compatibility with the 2 KHz rate selected for the coning algorithm.

In the case of the positioning algorithms discussed in Section 7.3, the "typical" form presented in Section 7.3.2 is usually adequate for most applications (to date). For the exceptional cases when very high resolution position updating is required, the time interval for the accuracy requirement is usually restricted to brief periods during the application mission profile. Moreover, for some of these applications, post-processing is acceptable using data recorded during the high resolution time interval, hence, the complexity of the high resolution algorithms would not be a real-time computer throughput issue. For example, for Synthetic Aperture Radar (SAR) motion compensation, high resolution position data is required for only brief intervals (e.g., 5 - 10 seconds) during SAR data acquisition, which may then be subsequently processed for SAR image formation. We also note that in high resolution applications, the earth referenced position of the INS chassis/mount is usually the required output, which equals the sum of earth referenced inertial sensor assembly position (calculated by the inertial navigation algorithms) plus vibration/specific-force induced displacement of the sensor assembly relative to the INS chassis/mount (due to compliance of elastomeric isolators that interface the sensor assembly to the INS chassis). Such displacement can be computed under dynamic maneuvers by appropriate digital filtering of vibration induced jitter (as in Chapter 9) and by quasi-static flexure modeling (i.e., displacement equals average negative specific force divided by the square of the sensor assembly isolator mount undamped natural frequency). This quasi-static correction can be seen from Equation (10.5.1-7) rearranged with $\ddot{x}$ identified as the specific force, $x - x_F$ identified as the sensor assembly displacement, $\frac{2\,k}{m}$ identified from (10.5.1-18) as the undamped natural frequency squared, $\dot{x} - \dot{x}_F$ approximated as zero under quasi-static conditions, and the δ terms ignored as negligible. Note that in principle, the displacement can also be measured directly using special sensing devices installed on the sensor assembly.

As an example of the inertial navigation position integration algorithm selection process, let us consider a high resolution application with an overall INS requirement for position error fluctuations to be "significantly less than 1 cm" during 5 - 10 second periods (not unusual for applications in which the actual requirement is a function of error frequency content and not clearly known). Allowing design margin for error in the sensor-assembly-to-chassis-mount flexure displacement calculation (described in the previous paragraph), we budget the INS accuracy specification into a requirement for the position algorithm to have less than 0.01 cm dynamic position error fluctuation during 5 - 10 sec. Let us further assume for this example that the basic position algorithm update rate has been selected to be 50 Hz and that the selected inertial velocity algorithm accuracy is compatible with high resolution position updating requirements (e.g., includes high rate sculling). Simplified "pencil-and-paper" analysis of the "typical" form (7.3.2-1) position algorithm (or other versions) can be used to assess its accuracy at 50 Hz using the high resolution algorithm to represent the correct truth model. An analytical model for the high resolution $\Delta \underline{R}_{SF_m}^{B(m-1)}$ increment truth model can be derived using $\Delta \underline{v}_{SF}^{B(m-1)}$ from (7.2.2.2-5) with (7.2.2.2-3) in (7.3.3-4):

$$\Delta \underline{R}_{SF_m}^{B(m-1)} = \int_{t_{m-1}}^{t_m} \Delta \underline{v}_{SF}^{B(m-1)}(\tau)\, d\tau$$

$$\Delta \underline{v}_{SF}^{B(m-1)}(\tau) = \int_{t_{m-1}}^{\tau} \left[I + \left(\underline{\alpha}(t) \times \right) \right] \underline{a}_{SF}^{B}\, dt \qquad \underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}^{B}\, d\tau \tag{7.4-1}$$

The Equation (7.3.2-1) "typical" algorithm is based on approximating the rate of change of velocity as constant. Neglecting the small L Frame rotation effect, this is equivalent to approximating $\Delta \underline{v}_{SF}^{B(m-1)}(\tau)$ in (7.4-1) as a linear function of time τ since t_{m-1} for which $\Delta \underline{R}_{SF_m}^{B(m-1)}$ in (7.3.3-4) would equal T_m times half the value of $\Delta \underline{v}_{SF}^{B(m-1)}(\tau)$ at $\tau = t_m$. Thus, position updating based on the (7.3.2-1) "typical" algorithm is equivalent to:

$$\Delta \underline{R}_{SF/Typ_m}^{B(m-1)} = \frac{1}{2} \Delta \underline{v}_{SF_m}^{B}\, T_m \qquad \Delta \underline{v}_{SF_m}^{B(m-1)} = \Delta \underline{v}_{SF}^{B(m-1)}(t_m) \tag{7.4-2}$$

where

$\Delta \underline{R}_{SF/Typ_m}^{B(m-1)}$ = "Typical" algorithm equivalent to $\Delta \underline{R}_{SF_m}^{B(m-1)}$.

For the (7.2.2.2.2-6) linearly ramping specific-force/angular-rate model in (7.4-1) - (7.4-2), the position increments for the truth model $\Delta\underline{R}_{SF_m}^{B(m-1)}$ and for the "typical" algorithm $\Delta\underline{R}_{SF/Typ_m}^{B(m-1)}$ become:

$$\Delta\underline{R}_{SF_m}^{B(m-1)} = \frac{1}{2}\,\underline{C}\,T_m^2 + \frac{1}{6}\left(\underline{D} + \underline{A}\times\underline{C}\right)T_m^3 + \frac{1}{12}\left(\underline{A}\times\underline{D} + \frac{1}{2}\,\underline{B}\times\underline{C}\right)T_m^4 + \frac{1}{40}\,\underline{B}\times\underline{D}\,T_m^5$$

$$\Delta\underline{R}_{SF/Typ_m}^{B(m-1)} = \frac{1}{2}\,\underline{C}\,T_m^2 + \frac{1}{4}\left(\underline{D} + \underline{A}\times\underline{C}\right)T_m^3 + \frac{1}{6}\left(\underline{A}\times\underline{D} + \frac{1}{2}\,\underline{B}\times\underline{C}\right)T_m^4 + \frac{1}{16}\,\underline{B}\times\underline{D}\,T_m^5 \tag{7.4-3}$$

Comparing $\Delta\underline{R}_{SF/Typ_m}^{B(m-1)}$ with the $\Delta\underline{R}_{SF_m}^{B(m-1)}$ truth model in (7.4-3) allows the error in $\Delta\underline{R}_{SF/Typ_m}^{B(m-1)}$ to be assessed for selected maneuver values. Under a constant specific force maneuver (i.e., $\underline{D} = 0$) with zero angular rate (i.e., $\underline{A} = \underline{B} = 0$), $\Delta\underline{R}_{SF/Typ_m}^{B(m-1)}$ equals $\Delta\underline{R}_{SF_m}^{B(m-1)}$, hence, is error free. Under a $\underline{D} = 3$ g/sec linear acceleration rate with $\underline{C} = 0$ and zero angular rate ($\underline{A} = \underline{B} = 0$), or for $\underline{A} = 1$ rad/sec angular rate with $\underline{C} = 3$ g acceleration and $\underline{B} = \underline{D} = 0$, the calculated error in $\Delta\underline{R}_{SF/Typ_m}^{B(m-1)}$ (using $T_m = 0.02$ sec for the 50 Hz update rate) is 0.00196 cm or $50 \times 0.00196 = 0.098$ cm in one second. Compared with the 0.01 cm in 5 - 10 sec requirement, the 0.098 cm in 1 sec figure would be considered unacceptable. Position algorithm assessment under vibration can also be analytically estimated. The velocity sampling process for the position integration algorithm can produce position error under vibration due to frequency folding effects (as described in Chapter 10, Section 10.1.3). For example, for the 3.6 g root-mean-square sensor assembly vibration (in the previous sculling example), the associated velocity vibration is 11.2 cm/sec root-mean-square centered around the sensor assembly 50 Hz mount resonance (which would be accurately measured by the hypothesized velocity algorithm). The aliasing (i.e., "folding" effect) error associated with sampling the vibrating velocity at 50 Hz for algorithm (7.3.2-1) can produce a $11.2 \times 0.02 = 0.22$ cm error each position update. If the error is random per update, the total cumulative error in 1 second (50 updates) would be $0.22 \times \sqrt{50} = 1.6$ cm; if the error is systematic, the position error in 1 second would be $0.22 \times 50 = 11$ cm. In either case, the algorithm error greatly exceeds the 0.01 cm over 5 - 10 sec requirement. A more sophisticated analytical assessment of position algorithm folding effect error can be made using Chapter 10, Equations (10.1.3.2.4-3), (10.3-20), (10.4.2-44), (10.6.1-25) or (10.6.2-21).

Based on such analyses, let us assume we have elected to use the (7.3.3-8) - (7.3.3-9) high resolution position algorithm to assure 5 - 10 sec 0.01 cm high quality resolution. The next question is which terms in (7.3.3-9) are to be included. The $\underline{S}_{\upsilon_m}$ term in (7.3.3-9) is the dominant term for integrating velocity into position and must be included. Under a 3 g constant specific force maneuver, $\underline{S}_{\upsilon_m}$ from (7.3.3-10) equals 0.59 cm per 50 Hz position update cycle or 29.4 cm in one second. The next most important term is the $\Delta\underline{R}_{Rot_m}$ position rotation compensation term. Using (7.3.3-11) with (7.3.3-10) input, the magnitude of $\Delta\underline{R}_{Rot_m}$ under a constant 3 g / 1 rad per sec maneuver is 0.0039 cm per update cycle or 0.20 cm cumulative position change in 1 second (Note, for a 3 g/sec linearly ramping specific force, $\underline{S}_{\upsilon_m}$ also equals 0.0039 cm per cycle and sums to 0.20 cm in one second). For the 0.01 cm requirement, the $\Delta\underline{R}_{Rot_m}$ term is, therefore, also needed. The question of whether to include the $\Delta\underline{R}_{Scrl_m}$ term can be addressed by analyzing the magnitude of $\Delta\underline{R}_{Scrl_m}$ under dynamic vibration motion using a rearranged version of (7.3.3-9):

$$\Delta\underline{R}_{Scrl_m} = \Delta\underline{R}_{SF_m}^{B(m-1)} - \underline{S}_{\upsilon_m} - \Delta\underline{R}_{Rot_m} \tag{7.4-4}$$

Consider the 3.6 g root-mean-square vibration condition under 1 rad/sec constant angular rate. For a 3.6 g root-mean-square pure sine wave (i.e., $3.6 \times \sqrt{2} = 5.1$ g amplitude) at the 50 Hz isolator resonance frequency, the magnitudes of $\underline{S}_{\upsilon_m}$ and $\underline{\upsilon}_m$ in 0.02 sec are, from (7.3.3-10), 0.32 cm and 0 cm/sec, respectively. For the 1 rad/sec angular rate over 0.02 sec, $\underline{\alpha}_m$ is 0.02 rad. Thus, from (7.3.3-11), $\Delta\underline{R}_{Rot_m}$ is $(0.32 \times 0.02) / 6 = 0.0011$ cm, which, if systematic, accumulates in one second to $0.0011 \times 50 = 0.053$ cm. If random from cycle to cycle, the error accumulation over 10 seconds would be $0.0011 \times \sqrt{50 \times 10} = 0.024$ cm. The true solution $\Delta\underline{R}_{SF_m}^{B(m-1)}$ for this particular case (pure sinusoidal acceleration with constant angular rate) can be found by noting, using analytical integration, that the double integral of $\left(\underline{\alpha}(t) \times\right) \underline{a}_{SF}^{B}$ in (7.4-1) is zero at the sinusoidal vibration cycle times. Thus, at the vibration cycle times and with the definition for $\underline{S}_{\upsilon_m}$ in (7.3.3-10), we see from (7.4-1) that $\Delta\underline{R}_{SF_m}^{B(m-1)} = \underline{S}_{\upsilon_m}$. Therefore, from (7.4-4), the cumulative magnitude of $\Delta\underline{R}_{Scrl_m}$ will equal the magnitude of $\Delta\underline{R}_{Rot_m}$, or from the previous numerical $\Delta\underline{R}_{Rot_m}$ assessment, $\Delta\underline{R}_{Scrl_m}$ will be 0.053 cm per second (if systematic) and 0.024 cm over 10 seconds (if random). To meet the accuracy requirement of 0.01 cm over 5 - 10 seconds, we conclude that $\Delta\underline{R}_{Scrl_m}$ will also be required.

The final question is which particular terms in $\Delta\underline{R}_{Scrl_m}$ algorithm (7.3.3-10) are needed. The answer can be obtained from similar individual analyses of each term in (7.3.3-10) to identify which are significant relative to the requirement. A simpler approach is to arbitrarily, but conservatively, use the full (7.3.3-10) form. The rationale might be that the savings in using a simplified version (e.g., without the second order terms) is not worth the time and cost for justification, assuming computer throughput is not an issue. This approach has additional merit because it frees the system designer of concern for INS algorithm error during the development optimization process of the system using the INS.

The position algorithm selection process described above is fairly rudimentary, admittedly conservative, but sufficient, if the outcome is the conservative approach of applying the full high resolution algorithm, particularly if the accuracy requirement cannot be clearly defined. Had the choice been to use the "typical" algorithm, or an alternate version thereof, a more sophisticated process would have been required to assure adequate performance over a more accurate and complete set of defined operating conditions. For example, complex maneuver/vibration profiles can be simulated and input to the "trial" algorithm, with its accuracy evaluated using the high resolution algorithm (with the same input) as a reference. In this regard, the high resolution algorithm can be viewed as a truth model for position algorithm evaluation, but available for use if the "trial" algorithm is inadequate. An assessment of the need to include particular terms in the scrolling portion of the high resolution algorithm can be made similarly; by calculating the magnitude of each term under simulated input versus error allowances (a term is needed if its magnitude exceeds the allowance). This step can be augmented using analytical models for input conditions, similar to the approach described in the previous example.

As pointed out in Reference 18, an interesting and often overlooked INS algorithm error mechanism that can be generated during initial self alignment is potential build-up from position integration algorithm folding effect error under vibration exposure. The self-alignment process frequently uses the integral of velocity (a position parameter known as "position divergence") for the input to the initial alignment Kalman filter (see Chapter 6, Section 6.1.2 and the Chapter 15 discussion in the second paragraph following Equation (15.2.1-1)). By processing the position divergence input, the Kalman filter is able to determine the horizontal components of earth rate in the local level L Frame from which initial INS heading is calculated. System effects that produce errors on the position divergence signal lead to earth rate estimation errors, hence, initial heading error. Two such effects are accelerometer horizontal quantization noise and ramping horizontal accelerometer error during the self alignment process. Similarly, folding effect error in the position divergence computation algorithm under vibration can also produce initial heading error. (The position divergence algorithm error has a dynamic time profile whose amplitude and frequency depend on the difference between the algorithm repetition rate and the frequency content of INS vibration during initial alignment - See Sections 10.1.3.2.4, 10.3, 10.4.2, 10.6.1, and 10.6.2 of Chapter 10). Sections 14.4 and 14.6.4.4 of Chapter 14 show how earth rate estimation error during initial self-alignment is impacted by the effect of accelerometer horizontal quantization noise and ramping horizontal error on the position divergence input to the initial alignment Kalman filter. Section 14.2 shows how earth rate estimation error creates initial heading error. For simplified analysis purposes, an equivalency can be defined between the effects of position divergence algorithm folding, accelerometer horizontal quantization noise and accelerometer ramping horizontal error on position divergence error. Through this equivalency and the above noted Chapter 14 section results, the effect of position divergence folding on initial heading error can be approximately evaluated. Using such results, guidelines can be established to assure that position divergence folding will not generate unacceptable initial heading error. The overall method is illustrated in Section 7.4.1 that follows.

So long as the selected integration algorithm is analytically valid, it can be improved in accuracy by increasing its repetition rate. All of the error mechanisms described in this section would be completely eliminated if their repetition rates were infinite. Continuing computer technology advances (increasing speed and decreasing program memory cost), therefore, tend to diminish any advantages one algorithm might have over another (usually measured primarily by accuracy for a given repetition rate and, secondarily, by required program memory). Excessively high repetition rates are to be avoided, however (even if computer throughput allowances permit) to limit error build-up caused by computer finite word length effects and rectification of high frequency multi-axis sensor errors (high frequency error output from one inertial sensor that is frequency correlated with outputs from sensors in the other axes; denoted as "pseudo-coning" error for the coning part of the attitude computation in Section 7.1 and "pseudo-sculling" error for the sculling part of the velocity calculation in Section 7.2). The finite computer word-length error effect is generally not a major factor with modern computer technology, typically having 64 bit double precision floating point word lengths. The pseudo-coning/sculling issue must be resolved on an individual design basis depending on the characteristics of high frequency error effects anticipated from the inertial sensor assembly in its operational dynamic environment. A general ground-rule to follow in coning/sculling algorithm repetition rate selection is to run the algorithms only as fast as required to accurately measure anticipated real multi-axis high frequency angular-rate/specific-force that can potentially rectify into real attitude/velocity change, but no faster, to minimize the likelihood of rectifying high frequency sensor output error into attitude/velocity error build-up.

The ultimate selection of algorithms to be used in a particular application is generally made based on prior experience of the responsible design engineer. The author has had long experience with the algorithms described in Sections 7.1 - 7.3 and feels comfortable adapting them to any strapdown application. They are well defined analytically, can be programmed using a simple sequential software executive structure, readily lend themselves to straightforward validation procedures, and are easily adapted to the requirements and constraints of particular applications.

7.4.1 ASSESSMENT OF POSITION INTEGRATION ALGORITHM FOLDING EFFECT ON INITIAL ALIGNMENT HEADING ERROR

This section describes an approximate method for assessing the effect of position integration algorithm folding on initial self-alignment heading error under sinusoidal and random vibration exposures. The method is based on results developed in Chapters 10 and 14, the appropriate sections of which the reader is encouraged to first thoroughly digest.

A rough assessment of initial alignment heading error due to position integration algorithm folding under <u>sinusoidal</u> vibration (at a particular frequency) can be made as follows. First, we identify $\delta \underline{R}_{SF/Algo}(t)$ in Sections 10.1.3.2.3 and 10.1.3.2.4 as the error in the position divergence computation algorithm, the input to the initial alignment Kalman filter as described in the previous section (third from last paragraph). Then we write for $\delta \underline{R}_{SF/Algo}(t)$ from Equations (10.1.3.2.4-1), (10.1.3.2.3-11) and (10.1.3.2.3-12) with t_M equated to the general running time in alignment t:

$$\begin{aligned} \delta \underline{R}_{SF/Algo}(t) \; = \; - \; \underline{u}_{Vib} \, \frac{1}{\Omega^2} \, a_{SF_0} \Bigg\{ \Bigg(& \frac{\Omega \, T_l \sin \Omega' \, T_l}{2 \, (1 - \cos \, \Omega' \, T_l)} \\ & + \frac{1}{12} \, \Omega \, T_l \sin \, \Omega' \, T_l - 1 \Bigg) \Big[\sin \Big(\Omega'(t - \, t_0) + \Omega \, t_0 - \varphi_{a_{SF}} \Big) - \sin \big(\Omega \, t_0 - \varphi_{a_{SF}} \big) \Big] \\ & - \frac{1}{12} \, \Omega \, T_l \Big[\cos \Big(\Omega'(t - \, t_0) + \Omega \, t_0 - \varphi_{a_{SF}} \Big) - \cos \big(\Omega \, t_0 - \varphi_{a_{SF}} \big) \Big] \, (1 - \cos \, \Omega' \, T_l) \Bigg\} \end{aligned} \tag{7.4.1-1}$$

where

T_l = Position divergence integration algorithm update time period.

a_{SF_0} = Sinusoidal vibration acceleration amplitude.

Ω = Sinusoidal vibration frequency.

k = Nearest integer value of the ratio of Ω to $2 \, \pi \, / \, T_l$.

Ω' = Folded frequency defined analytically as $\Omega - \dfrac{2 \, \pi \, k}{T_l}$.

The effect of the (7.4.1-1) position divergence error on initial heading alignment performance can be estimated from its third derivative (i.e., time ramping acceleration) which, due to the sinusoidal terms, acts similar to an oscillating accelerometer bias trending error on the position divergence input to the initial alignment Kalman filter (see previous section - third from last paragraph). The third derivative of (7.4.1-1) is:

$$\begin{aligned} \delta \dddot{\underline{R}}_{SF/Algo}(t) \; = \; \underline{u}_{Vib} \, \frac{1}{\Omega^2} \, a_{SF_0} \, \big(\Omega' \big)^3 \Bigg\{ \Bigg(& \frac{\Omega \, T_l \sin \, \Omega' \, T_l}{2 \, (1 - \cos \, \Omega' \, T_l)} \\ & + \frac{1}{12} \, \Omega \, T_l \sin \, \Omega' \, T_l - 1 \Bigg) \cos \Big(\Omega'(t - \, t_0) + \Omega \, t_0 - \varphi_{a_{SF}} \Big) \\ & + \frac{1}{12} \, \Omega \, T_l \Big[\sin \Big(\Omega'(t - \, t_0) + \Omega \, t_0 - \varphi_{a_{SF}} \Big) \Big] \, (1 - \cos \, \Omega' \, T_l) \Bigg\} \end{aligned} \tag{7.4.1-2}$$

For $k > 0$ and for small values of $\Omega' \, T_l$, Equation (10.1.3.2.3-14) shows that $\dfrac{\Omega \, T_l \sin \Omega' \, T_l}{2 \, (1 - \cos \, \Omega' \, T_l)} \approx \dfrac{2 \, \pi \, k}{\Omega' \, T_l}$. For small $\Omega' \, T_l$ (compared to one) we can also approximate $\Omega = \dfrac{2 \, \pi \, k}{T_l}$. Thus, for $\dfrac{1}{\Omega' \, T_l}$ large compared to $\Omega \, T_l$ (and to one), (7.4.1-2) can be approximated as:

$$
\begin{aligned}
\delta \dddot{\underline{R}}_{SF/Algo}(t) &\approx \underline{u}_{Vib} \, \frac{1}{\left(\dfrac{2 \, \pi \, k}{T_l}\right)^2} \, a_{SF_0} \left(\Omega'\right)^3 \frac{2 \, \pi \, k}{\Omega' \, T_l} \cos\left(\Omega'(t - t_0) + \Omega \, t_0 - \varphi_{a_{SF}}\right) \\
&= \underline{u}_{Vib} \, a_{SF_0} \left(\Omega'\right)^2 \frac{T_l}{2 \, \pi \, k} \cos\left(\Omega'(t - t_0) + \Omega \, t_0 - \varphi_{a_{SF}}\right)
\end{aligned}
\qquad (7.4.1\text{-}3)
$$

Equation (7.4.1-3) allows the third derivative of the position divergence algorithm error to be evaluated as a function of the vibration amplitude, vibration frequency (as a multiple k of the algorithm update frequency $\dfrac{2 \, \pi}{T_l}$), folding frequency and algorithm update time period. The effect of (7.4.1-3) on initial alignment accuracy is very pronounced when the folding frequency Ω' is on the order of the alignment time so that the (7.4.1-3) effect on alignment behaves as a "steady" acceleration trending over the alignment period (similar to a steady accelerometer horizontal bias trending error). On the other hand, the amplitude of (7.4.1-3) increases with folding frequency. However, the author knows from experience that for large Ω' (e.g., greater than $2 \, \pi \times 10$ Hz), the folding effect, even though large in amplitude, generally has no serious impact on initial alignment heading determination accuracy because the alignment Kalman filter effectively attenuates dynamic inputs in this frequency range. Let us now look at the folding effect for the case when the cycle time period associated with Ω' (i.e., $\dfrac{2 \, \pi}{\Omega'}$) equals the alignment time (or $\Omega' = \dfrac{2 \, \pi}{T_{Align}}$), and the (7.4.1-3) amplitude becomes:

$$
\delta \dddot{R}_{SF/Algo_{Amp}} \approx a_{SF_0} \left(\frac{2 \, \pi}{T_{Align}}\right)^2 \frac{T_l}{2 \, \pi \, k} = a_{SF_0} \, \frac{2 \, \pi \, T_l}{k \, T_{Align}^2} \qquad (7.4.1\text{-}4)
$$

where

T_{Align} = Alignment time.

If we treat (7.4.1-4) as the equivalent to an accelerometer bias trending, we can use Equation (14.4-6) to estimate the resulting initial heading error as $\delta \dddot{R}_{SF/Algo_{Amp}}$ divided by the product of gravity magnitude with horizontal earth rate. As a numerical example, consider the case in which $a_{SF_0} = 1$ g, $T_l = 0.02$ sec, $T_{Align} = 300$ sec and $k = 1$. Then at 45 deg latitude the estimated folding effect heading error ψ would be:

$$\psi \approx \frac{2\,\pi \times 0.02}{300^2} \frac{1}{\dfrac{15 \text{ deg/hr}}{57.3 \times 3600 \text{ rad/sec per deg/hr}} \cos 45} 10^3 \text{ mil-rad/rad} = 27.2 \text{ mil-rad}$$

Clearly this would be a disastrous situation for a standard accuracy aircraft type INS for which the initial heading alignment accuracy requirement is typically on the order of 1 mil-rad. In order to eliminate this error effect, assurance must be provided that any discrete vibration frequency that may exist on the sensor assembly during initial alignment will not be an integer multiple of the position algorithm update frequency (a classical example of such a discrete vibration effect is the linear acceleration induced in strapdown accelerometers from back reaction torque transmitted into the sensor assembly by mechanically dithered laser gyros (Reference 32)). This can be achieved by running the algorithm faster than the highest frequency expected, or by controlling the frequency source itself (if possible) to be far enough removed from the algorithm update frequency. Sufficient safety margin must be provided in either case. The degree of margin required should be assessed using a simulation of the alignment process in the presence of discrete sinusoidal frequency vibration input (e.g., a single horizontal axis version of Equations (15.2.1.1-21) with simulated horizontal sinusoidal acceleration vibration used for $\left(C_B^N\, \underline{a}_{SF}^B\right)_H$).

An estimate for the initial alignment heading error due to position integration folding under <u>random</u> vibration can be made by first calculating the position divergence algorithm error variance ($\mathcal{E}\left(\delta R_{SF/Algo}^2(t)\right)$ in Equations (10.6.1-25)) for t equal to the alignment time. The $\mathcal{E}\left(\delta R_{SF/Algo}^2(t)\right)$ expression in Equations (10.6.1-25) has a dynamic time profile whose amplitude and frequency depend on the difference between the algorithm repetition rate and the frequency content of the input vibration. From Equations (12.5.1-1) we see that accelerometer quantization error $\delta\underline{\upsilon}_{Quant}$ integrates directly into position error $\delta\underline{R}^N$. Therefore, from Equation (15.1.2.1.1-30), accelerometer quantization error generates a build-up in the position divergence error variance equal to the quantization noise density $q_{\upsilon Quant}$ multiplied by the velocity-into-position integration time t. We can then calculate the equivalent $q_{\upsilon Quant}$ value that would produce the same position error variance as the position integration algorithm folding effect by dividing $\mathcal{E}\left(\delta R_{SF/Algo}^2(t)\right)$ by t. With $q_{\upsilon Quant}$ so determined, the associated earth rate estimation error during alignment $P_{\Omega_{\upsilon Quant}}$ is calculated as $\dfrac{20\, q_{\upsilon Quant}}{g^2\, t^5}$ (from Equations (14.6.4.4-2)).

The resulting initial heading error is the square root of $P_{\Omega_{\upsilon}\mathrm{Quant}}$ divided by horizontal earth rate. A more precise assessment of the effect of sampling on initial alignment accuracy can be made by simulating the alignment process equations using a simulated random vibration signature as the measurement noise (e.g., a single horizontal axis version of Equations (15.2.1.1-21) with simulated horizontal random acceleration vibration used for $\left(C_B^N \, \underline{a}_{SF}^B\right)_H$. The acceleration vibration can be simulated as white acceleration noise feeding a second order spring/damper system such as Equations (15.2.1.2-8)).

7.5 STRAPDOWN INERTIAL NAVIGATION SYSTEM ALGORITHM SUMMARY

Table 7.5-1 is a listing of the principal algorithms from Chapter 7 (including Chapter 4 output equations and Chapter 5, Table 5.6-1 earth related parameters) typically utilized in strapdown inertial navigation system software packages. Table 7.5-1 lists the algorithm function, input parameters, output parameters and equation number. Definitions for the input/output parameters are provided in the sections in which the algorithms/equations are derived as identified in the Parameter Index at the end of the book.

Table 7.5-1 Summary Of Typical Strapdown Inertial Navigation System Computational Algorithms

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
HIGH SPEED CALCULATIONS			
Integrated B Frame Angular Rate Increments	$\Delta\underline{\alpha}_l$	$\underline{\alpha}_l$, $\underline{\alpha}_m$	(7.1.1.1.1-17)
Integrated B Frame Acceleration Increments	$\Delta\underline{\upsilon}_l$	$\underline{\upsilon}_l$, $\underline{\upsilon}_m$	(7.2.2.2.2-14)
Coning Increment	$\Delta\underline{\alpha}_l$, $\underline{\alpha}_l$	$\underline{\beta}_m$	(7.1.1.1.1-18)
Sculling Increment	$\Delta\underline{\alpha}_l$, $\underline{\alpha}_l$, $\Delta\underline{\upsilon}_l$, $\underline{\upsilon}_l$	$\Delta\underline{v}_{Scul_l}$, $\Delta\underline{v}_{Scul_m}$	(7.2.2.2.2-15)
High Speed Update Time Interval (Constant)	Constant	T_l	—

(Continued)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Doubly Integrated B Frame Angular Rate And Acceleration Increments (For High Resolution Position Algorithm)	$\Delta\underline{\alpha}_l$, $\underline{\alpha}_l$, $\Delta\underline{\upsilon}_l$, $\underline{\upsilon}_l$, T_l	$\Delta\underline{S}_{\alpha_l}$, $\underline{S}_{\alpha_l}$, $\underline{S}_{\alpha_m}$, $\Delta\underline{S}_{\upsilon_l}$, $\underline{S}_{\upsilon_l}$, $\underline{S}_{\upsilon_m}$	(7.3.3.2-18), (7.3.3.2-19)
Scrolling Increment (For High Resolution Position Algorithm)	$\Delta\underline{\alpha}_l$, $\underline{\alpha}_l$, $\Delta\underline{S}_{\alpha_l}$, $\underline{S}_{\alpha_l}$, $\Delta\underline{\upsilon}_l$, $\underline{\upsilon}_l$, $\Delta\underline{S}_{\upsilon_l}$, $\underline{S}_{\upsilon_l}$, $\Delta\underline{v}_{Scul_l}$, T_l	$\Delta\underline{R}_{Scrl_m}$	(7.3.3.2-20)

NORMAL SPEED CALCULATIONS

EARTH RELATED PARAMETERS

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Earth Shape, Gravity And Angular Rate Constants	Reference 4	R_0, e, μ, J_2, J_3, .., ω_e	Reference 4
Earth Polar Axis Component of Geodetic Vertical Unit Vector	C_N^E	$u_{Up_{YE}}$	(5.3-16)
Modified Radial Distance to Earth Surface Location	R_0, e, $u_{Up_{YE}}$	R_S'	(5.1-10)
Radial Distance to Earth Surface Location	R_S', e, $u_{Up_{YE}}$	R_S	(5.2.1-4)
Radial Distance to Navigation Point	R_0, R_s, R_S', h	R	(5.2.1-5)
Cosine Of Range Vector Polar Coordinate Angle	$u_{Up_{YE}}$, e, R, h, R_S'	$\cos\phi$	(5.2.2-3)
Modified Sine Of Range Vector Polar Coordinate Angle	R, h, R_S'	$\left(\dfrac{\sin\phi}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$	(5.2.2-5)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Cosine And Modified Sine Of Difference Between Geocentric And Geodetic Latitudes	$R, h, R_S', e, u_{Up_{YE}}$	$\cos \partial l, \left(\frac{\sin \partial l}{\sqrt{1 - u_{Up_{YE}}^2}} \right)$	(5.2.3-5)
Local Earth Surface Point Radius Of Curvature In Latitude Direction	R_0, e, R_S'	r_{ls}	(5.2.4-25)
Local Navigation Point Radius Of Curvature In Latitude Direction	r_{ls}, h	r_l	(5.2.4-37)
Curvature Matrix In The N Frame	r_l, C_N^E, h, R_S', e	F_C^N	(5.3-18)
Vertical Transport Rate Component	Section 4.5 For Options	ρ_{ZN}	Section 4.5 For Options
Unit Vector Upward In N Frame	Definition	$\underline{u}_{ZN}^N$	(5.3-18)
N Frame Transport Rate Vector	$\underline{v}^N, F_C^N, \rho_{ZN}, \underline{u}_{ZN}^N$	$\underline{\rho}^N = \underline{\omega}_{EN}^N$	(5.3-17)
Gravity Components In Polar Coordinates	$\mu, R, R_0, \cos \phi, J_2, J_3, \ldots$	$g_r, \left(\frac{g_\phi}{\sin \phi} \right), g_\theta$	(5.4-1) & (5.4-2)
North And Vertical Gravity Components	$g_r, \left(\frac{g_\phi}{\sin \phi} \right), u_{Up_{YE}}, \cos \partial l, \left(\frac{\sin \partial l}{\sqrt{1 - u_{Up_{YE}}^2}} \right), \left(\frac{\sin \phi}{\sqrt{1 - u_{Up_{YE}}^2}} \right)$	$\left(\frac{g_{North}}{\sqrt{1 - u_{Up_{YE}}^2}} \right), g_{Up}$	(5.4-4)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
North And Vertical Plumb-bob Gravity Components	$\left(\frac{g_{North}}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$, g_{Up}, R_S', h, $u_{Up_{YE}}$, ω_e	$\left(\frac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$, $g_{P_{Up}}$	(5.4.1-9)
N Frame Plumb-bob Gravity Components	$\left(\frac{g_{P_{North}}}{\sqrt{1 - u_{Up_{YE}}^2}}\right)$, $g_{P_{Up}}$, C_N^E	$\underline{g}_P^N$	(5.4.1-11)
N Frame Earth Rate Vector	C_N^E, ω_e	$\underline{\omega}_{IE}^N$	(4.1.1-3), (4.1.1-4)
<u>VELOCITY CALCULATIONS</u>			
B Frame Velocity Rotation Compensation (Exact Formulation)	$\underline{\alpha}_m$, $\underline{\upsilon}_m$	$\Delta\underline{v}_{Rot_m}$	(7.2.2.2.1-7), (7.2.2.2.1-8)
B Frame Velocity Rotation Compensation (First Order Approximation Form)	$\underline{\alpha}_m$, $\underline{\upsilon}_m$	$\Delta\underline{v}_{Rot_m}$	(7.2.2.2-25)
B Frame Integrated Specific Force Acceleration Increment	$\underline{\upsilon}_m$, $\Delta\underline{v}_{Rot_m}$, $\Delta\underline{v}_{Scul_m}$	$\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$	(7.2.2.2-23)
L Frame Integrated Specific Force Acceleration Increment	$\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$, $C_{B_{I(m-1)}}^{L_{I(n-1)}}$	$\Delta\underline{v}_{SF_m}^{L_{I(n-1)}}$	(7.2.2-2)
N To L Frame Direction Cosine Matrix (Constant)	Definition	$C_N^L = \left(C_L^N\right)^T$	(4.1.1-2)
Velocity (And Attitude) m Cycle Update Time Interval, m Cycles Per n Cycle, And m Cycles Since n-1	Constants	T_m, j, r	—

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
L Frame Rotation Vector (Cycle n-1 To m)	$\underline{\omega}_{IE}^{N}$, ρ_{ZN}, F_C^N, $\underline{v}^N$, T_m, r, j, C_N^L, $\underline{u}_{ZN}^N$	$\underline{\zeta}_{n-1,m}$	(7.2.2.1-4) - (7.2.2.1-7)
L Frame Rotation Matrix (First Order Form)	$\underline{\zeta}_{n-1,m}$	$C_{L_{I(n-1)}}^{L_{I(m)}} = C_{L_{(n-1)}}^{L_{(m)}}$	(7.2.2.1-1)
L Frame Rotation Compensation	$\Delta\underline{v}_{SF_m}^{L_{I(n-1)}}$, $C_{L_{I(n-1)}}^{L_{I(m)}}$	$\Delta\underline{v}_{SF_m}^{L}$	(7.2.2-4)
Integrated Coriolis Acceleration & Plumb-bob Gravity Increment	$\underline{g}_P^N$, $\underline{\omega}_{IE}^N$, ρ_{ZN}, F_C^N, $\underline{v}^N$, $\underline{u}_{ZN}^N$, T_m, r, j	$\Delta\underline{v}_{G/COR_m}^{N}$	(7.2.1-1) - (7.2.1-3)
N Frame Velocity Update	$\Delta\underline{v}_{SF_m}^{L}$, $\Delta\underline{v}_{G/COR_m}^{N}$, C_L^N, $\underline{v}_{m-1}^N$	$\underline{v}_m^N$	(7.2-2)
Vertical Channel Control Gains	Constants	C_1, C_2, C_3	(4.4.1.2.1-11)
Altitude (And Position) Update Time Interval	Constant	T_n	—
Vertical Channel Control Signals	C_1, C_2, C_3, T_n, h_{Prsr_n}, h_n	e_{vc1_n}, e_{vc2_n}	(7.2-6)
Vertical Velocity Control	$\underline{v}_{n-}^N$, $\underline{u}_{ZN}^N$, e_{vc1_n}, T_n	$\underline{v}^N$	(7.2-5)
East, North, Up Velocity Component Outputs	$\underline{v}^N$, α	v_{East}, v_{North}, v_{Up}	(4.3.1-4)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
POSITION CALCULATIONS			
Position Rotation Compensation (High Resolution Position Algorithm - Exact Form)	$\underline{\alpha}_m$, $\underline{S}_{\alpha_m}$, $\underline{\upsilon}_m$, $\underline{S}_{\upsilon_m}$	$\Delta\underline{R}_{Rot_m}$	(7.3.3.1-16)
Position Rotation Compensation (High Resolution Position Algorithm - First Order Accuracy Form)	$\underline{\alpha}_m$, $\underline{S}_{\alpha_m}$, $\underline{\upsilon}_m$, $\underline{S}_{\upsilon_m}$	$\Delta\underline{R}_{Rot_m}$	(7.3.3-11)
Body Frame Position Increment Due To Specific Force Acceleration (High Resolution Position Algorithm)	$\underline{S}_{\upsilon_m}$, $\Delta\underline{R}_{Rot_m}$, $\Delta\underline{R}_{Scrl_m}$	$\Delta\underline{R}^B_{SF_m}$	(7.3.3-9)
N Frame Position Increment (High Resolution Position Algorithm)	$\underline{v}^N_{m-1}$, $\Delta\underline{v}^N_{G/COR_m}$, $\Delta\underline{R}^B_{SF_m}$, $\Delta\underline{v}^{L(n-1)}_{SF_m}$, $C^{L(n-1)}_{B(m-1)} = C^{LI(n-1)}_{BI(m-1)}$, $\underline{\zeta}_{n-1,m}$, $\underline{\zeta}_{n-1,m-1}$, $C^{L(m-1)}_{L(n-1)} = C^{LI(m-1)}_{LI(n-1)}$, C^N_L, T_m	$\Delta\underline{R}^N_m$	(7.3.3-8)
N Frame Position Increment (Trapezoidal Position Algorithm)	$\underline{v}^N$, T_m	$\Delta\underline{R}^N_m$	(7.3.2-1)
Altitude Change	$\Delta\underline{R}^N_m$, $\underline{u}^N_{ZN}$	Δh_n	(7.3.1-3)
Position Rotation Vector	ρ_{ZN}, F^N_C, $\Delta\underline{R}^N_m$, $\underline{u}^N_{ZN}$, T_n	$\underline{\xi}_n$	(7.3.1-11), (7.3.1-12)
Position Rotation Change Matrix	$\underline{\xi}_n$	$C^{NE(n-1)}_{NE(n)}$	(7.3.1-8)
Altitude Update	h_{n-1}, Δh_n	h_n	(7.3.1-1)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Altitude Control	h_{n-}, e_{vc2_n}, T_n	h_{n+}	(7.3.1-5)
Position Direction Cosine Matrix Update	$C^{E}_{N_{E(n-1)}}$, $C^{N_{E(n-1)}}_{N_{E(n)}}$	$C^{E}_{N_{E(n)}}$	(7.3.1-6)
Latitude, Longitude Outputs And Wander Angle	C^{E}_{N}	l, L, α	(4.4.2.1-3)
<u>ATTITUDE CALCULATIONS</u>			
B Frame Rotation Vector	$\underline{\alpha}_m$, $\underline{\beta}_m$	$\underline{\phi}_m$	(7.1.1.1-12)
B Frame Rotation Matrix (For Attitude Direction Cosine Matrix Updating)	$\underline{\phi}_m$	$C^{B_{I(m-1)}}_{B_{I(m)}}$	(7.1.1.1-3)
B Frame Rotation Quaternion (For Attitude Quaternion Updating)	$\underline{\phi}_m$	$q^{B_{I(m-1)}}_{B_{I(m)}}$	(7.1.2.1-3)
Attitude Update For B Frame Rotation (Direction Cosine Matrix Form)	$C^{L_{I(n-1)}}_{B_{I(m-1)}}$, $C^{B_{I(m-1)}}_{B_{I(m)}}$	$C^{L_{I(n-1)}}_{B_{I(m)}}$	(7.1.1.1-1)
Attitude Update For B Frame Rotation (Quaternion Form)	$q^{L_{I(n-1)}}_{B_{I(m-1)}}$, $q^{B_{I(m-1)}}_{B_{I(m)}}$	$q^{L_{I(n-1)}}_{B_{I(m)}}$	(7.1.2.1-1)
Position And L Frame Update Time Interval	Constant	T_n	—
L Frame Rotation Vector (Cycle n-1 To n)	$\underline{\omega}^{N}_{IE}$, ρ_{ZN}, F^{N}_{C}, $\Delta\underline{R}^{N}_{m}$, $\underline{u}^{N}_{ZN}$, T_n, C^{L}_{N}	$\underline{\zeta}_n$	(7.1.1.2.1-3), (7.1.1.2.1-5)
L Frame Rotation Matrix For Attitude Direction Cosine Matrix Updating (Exact Form)	$\underline{\zeta}_n$	$C^{L_{I(n)}}_{L_{I(n-1)}}$	(7.1.1.2-3)

ALGORITHM FUNCTION	**INPUT**	**OUTPUT**	**EQUATION**
L Frame Quaternion For Attitude Quaternion Updating (Exact Form)	$\underline{\zeta}_n$	$q_{L_{I(n-1)}}^{L_{I(n)}}$	(7.1.2.2-3)
Attitude Update For L Frame Rotation (Direction Cosine Matrix Form)	$C_{B_{I(m)}}^{L_{I(n-1)}}$, $C_{L_{I(n-1)}}^{L_{I(n)}}$	$C_{B_{I(m)}}^{L_{I(n)}}$	(7.1.1.2-1)
Attitude Update For L Frame Rotation (Quaternion Form)	$q_{B_{I(m)}}^{L_{I(n-1)}}$, $q_{L_{I(n-1)}}^{L_{I(n)}}$	$q_{B_{I(m)}}^{L_{I(n)}}$	(7.1.2.2-1)
Normalization And Orthogonalization Corrections (For Attitude Direction Cosine Matrix)	C_B^L	C_B^L	(7.1.1.3-1), (7.1.1.3-10)
Normalization Corrections (For Attitude Quaternion)	q_B^L	q_B^L	(7.1.2.3-6), (7.1.2.3-8)
Attitude Quaternion To Attitude Direction Cosine Matrix Conversion (For Attitude Quaternion As Basic Attitude Form)	q_B^L	C_B^L	(7.1.2.4-1)
Roll, Pitch, True Heading Euler Angle Outputs	C_B^L, α	ϕ, θ, ψ_{True}	(4.1.2-1), (4.1.2-2)

8 Navigation System Component Compensation Algorithms

8.0 OVERVIEW

In manufacturing an inertial navigation system it is virtually impossible to control tolerances so that the component accuracies are consistent with system navigational accuracy requirements. However, manufactured component inaccuracies are generally stable and analytically predictable, which provides the basis for correcting the inaccuracies as a system software operation. Once the component error effects are "characterized" (or "modeled") analytically (a combined analytical/component test iterative design process), system software algorithms are designed to compensate the component errors as part of the navigation system software computational process. The error effects for each system component are then measured as part of manufacturing/test operations (the "calibration" process) and the resulting measurement data (in the form of "calibration coefficients") are installed in the system software for operation with the component compensation algorithms. The residual errors left after system software compensation (e.g., due to component instabilities, calibration error, unmodelable error effects, and calibration algorithm error) then determines the ultimate accuracy of the inertial navigation system.

This chapter describes error models and compensation algorithms that can be used to correct for errors in the strapdown inertial sensors (angular rate sensors and accelerometers), relative displacement between accelerometers ("size effect"), misalignment of the strapdown sensor assembly relative to the system mount, and alignment of the system mount in the user vehicle relative to vehicle reference axes. Included is a discussion of the application of the sensor compensation algorithms to the Chapters 7 and 19 (Section 19.1) strapdown inertial navigation integration routines and their associated coning, sculling and scrolling elements. A table is provided at the end of the chapter listing the principal calibration equations developed in the order they would be processed and applied in the strapdown navigation computer.

Depending on his/her experience in the development and application of inertial sensor compensation algorithms, the reader may find the forms presented in this chapter to be more complex than the usual linearized treatment of error effects. In keeping with the basic theme of the book, the forms presented here are provided whenever possible without resorting to simplifying approximations (unless that is the only practical solution). The reader should recognize, however, that when operating the presented compensation algorithms in system software, the more complicated calculations shown for the compensation coefficients can be either pre-computed or executed once at system start-up (if not time varying), or executed at a low speed background computation cycle rate (if time varying). The reader should also recognize that many of the matrix operations shown are performed with diagonal matrices that have two simplifying properties: 1. Products of diagonal matrices are diagonal with each element equal to the product of the corresponding elements in the multiplying matrices, and 2. The inverse of a diagonal matrix is diagonal with each element equal to the reciprocal of the corresponding element in the original matrix. These observations considerably simplify the system software execution rate requirements for many of the compensation equations presented.

The principal coordinate frame used in this chapter is the sensor "body" B Frame as defined in Section 2.2.

8.1 INERTIAL SENSOR COMPENSATION ALGORITHMS

This section derives representative algorithms for compensating the angular rate sensor and accelerometer signals in a strapdown inertial navigation system. Both "Sensor Level" and "System Level" compensation terms are described; Sensor Level compensation is measured at the component level (angular rate sensor or accelerometer), System Level compensation is measured after system assembly; both are applied as part of the system computational process to correct the data used to update attitude, velocity, and position. Also described is the analytical process for calculating the sensor compensation coefficients from sensor measurements and the Chapter 18 System Level test results.

8.1.1 INERTIAL SENSOR ERROR CHARACTERISTICS AND COMPENSATION FORMULAS

This section characterizes the errors typically present in the raw inertial sensor outputs (angular rate sensors and accelerometers) and then derives a general form of compensation equations for correcting the errors.

8.1.1.1 ANGULAR RATE SENSOR ERROR CHARACTERISTICS AND COMPENSATION FORMULAS

The output vector from a strapdown angular rate sensor triad can be characterized as a function of its input angular rate vector as:

$$\underline{\omega}_{Puls} = \frac{1}{\Omega_{Wt_0}} \left(I + F_{Scal} \right) \left(F_{Algn}\, \underline{\omega} + \delta\underline{\omega}_{Bias} + \delta\underline{\omega}_{Quant} + \delta\underline{\omega}_{Rand} \right) \tag{8.1.1.1-1}$$

where

$\underline{\omega}$ = Angular rate vector sensed by the angular rate sensor triad. The components of $\underline{\omega}$ are the angular rate vector projections onto the sensor (or "Body") B Frame.

$\underline{\omega}_{Puls}$ = Angular rate sensor triad output vector in pulses per second. Each axis output pulse is a digital indication that the sensor associated with that axis has rotated through an integrated angular rate increment around its input axis equal to that particular sensor's pulse size.

Ω_{Wt_0} = Nominal pulse weight (a positive value) for each angular rate sensor (radians per pulse).

I = Identity matrix.

F_{Scal} = Angular rate sensor triad scale factor correction matrix; a diagonal matrix in which each element adjusts the output pulse scaling to correspond to the actual scaling for the particular sensor output. Nominally, the F_{Scal} matrix is zero. The F_{Scal} matrix may include non-linear scale factor effects and temperature dependency.

F_{Algn} = Alignment matrix for the angular rate sensor triad. Each row represents a unit vector along a particular angular rate sensor input axis as projected onto the B-Frame. Nominally, the F_{Algn} matrix is identity. The F_{Algn} matrix may include specific force acceleration dependency.

$\delta\underline{\omega}_{Bias}$ = Angular rate sensor triad bias vector. Each element equals the systematic output from a particular angular rate sensor under zero input angular rate conditions. In some angular rate sensors, $\delta\underline{\omega}_{Bias}$ may have environmental sensitivities (e.g., temperature and specific force acceleration dependency).

$\delta\underline{\omega}_{Quant}$ = Instantaneous angular rate sensor triad pulse quantization error associated with the output only being provided when the cumulative input equals the pulse weight per axis. Includes pulse output logic dead-band effect under turn-around conditions (see Section 8.1.3.2).

$\delta\underline{\omega}_{Rand}$ = Angular rate sensor triad random error output vector.

Equation (8.1.1.1-1) can be solved for the B Frame angular rate input vector as follows:

$$\underline{\omega}' = \Omega_{Wt_0}\left(I + F_{Scal}\right)^{-1} \underline{\omega}_{Puls} \qquad (8.1.1.1\text{-}2)$$

$$\underline{\omega} = F_{Algn}^{-1}\left(\underline{\omega}' - \delta\underline{\omega}_{Bias} - \delta\underline{\omega}_{Quant} - \delta\underline{\omega}_{Rand}\right) \qquad (8.1.1.1\text{-}3)$$

where

$\underline{\omega}'$ = Scale factor compensated angular rate sensor output vector.

Equation (8.1.1.1-2) represents the scale factor compensation equation for the raw angular rate sensor triad $\underline{\omega}_{Puls}$ output. Compensation for the remaining predictable errors in $\underline{\omega}_{Puls}$ is achieved using a simplified form of (8.1.1.1-3) in which it is recognized that the $\delta\underline{\omega}_{Rand}$ component is unpredictable, hence, can only be approximated by zero:

$$\underline{\omega} \approx F_{Algn}^{-1}\left(\underline{\omega}' - \delta\underline{\omega}_{Bias} - \delta\underline{\omega}_{Quant}\right) \tag{8.1.1.1-4}$$

Compensation Equations (8.1.1.1-2) and (8.1.1.1-4) are further refined to a more familiar form by introducing the following definitions:

$$\Omega_{Wt} \equiv \Omega_{Wt_0}\left(I + F_{Scal}\right)^{-1} \tag{8.1.1.1-5}$$

$$K_{Mis} \equiv I - F_{Algn}^{-1} \tag{8.1.1.1-6}$$

$$\underline{K}_{Bias} \equiv F_{Algn}^{-1}\,\delta\underline{\omega}_{Bias} \tag{8.1.1.1-7}$$

where

Ω_{Wt} = Angular rate sensor triad scale factor pulse weighting matrix (radians per pulse).

K_{Mis} = Angular rate sensor triad misalignment compensation matrix (nominally zero).

$\underline{K}_{Bias}$ = Angular rate sensor bias compensation vector.

Substituting (8.1.1.1-5) - (8.1.1.1-7) into (8.1.1.1-2) and (8.1.1.1-4) then obtains the equivalent compensation equations:

$$\begin{aligned} \underline{\omega}' &= \Omega_{Wt}\,\underline{\omega}_{Puls} \\ \underline{\omega} &\approx \underline{\omega}' - K_{Mis}\,\underline{\omega}' - \underline{K}_{Bias} - F_{Algn}^{-1}\,\delta\underline{\omega}_{Quant} \end{aligned} \tag{8.1.1.1-8}$$

Equations (8.1.1.1-5) - (8.1.1.1-8) constitute the compensation equations for the angular rate sensor output vector in continuous angular rate vector format. These equations will form the basis for the angular rate sensor triad compensation algorithms presented in Section 8.1.2.1.

8.1.1.1.1 Sensor And System Level Compensation Coefficient Evaluation For The Angular Rate Sensors

Compensation of the $\underline{\omega}_{Puls}$ angular rate sensor triad output as described in Section 8.1.1.1 requires that F_{Scal}, F_{Algn} and $\delta\underline{\omega}_{Bias}$ be evaluated for the angular rate sensors. This typically involves use of generic analytic algorithms that characterize sensor performance in terms of coefficients that are unique to each individual sensor. Determination of the coefficients is the calibration process performed as part of manufacturing/assembly/test operations. Calibration coefficient updates may also be performed during the INS operational life to sustain acceptable performance. A typical "factory" calibration process consists of two steps: "Sensor Level" calibration and "System Level" calibration. Sensor Level calibration is performed on each individual sensor before installation into the strapdown sensor assembly, and typically determines (for the angular rate sensors) the primary contributors to F_{Scal} and $\delta\underline{\omega}_{Bias}$. Following Sensor Level calibration, the sensors are installed in the sensor assembly and System Level calibration is performed. System Level calibration determines (for the angular rate sensors) the F_{Algn} matrix and residual corrections to F_{Scal} and $\delta\underline{\omega}_{Bias}$.

System Level calibration is performed on a sensor assembly that is being compensated using Sensor Level compensation equations. Hence, in effect, the following forms of Equations (8.1.1.1-2) and (8.1.1.1-4) apply for the angular rate sensor outputs during the System Level calibration process:

$$\underline{\omega}'^{*} = \Omega_{Wt_0}\left(I + F_{SensScal}\right)^{-1} \underline{\omega}_{Puls} \qquad (8.1.1.1.1\text{-}1)$$

$$\underline{\omega}^{*} = F_{SensAlgn}^{-1}\left(\underline{\omega}'^{*} - \delta\underline{\omega}_{SensBias} - \delta\underline{\omega}^{*}_{Quant}\right) \qquad (8.1.1.1.1\text{-}2)$$

where

$F_{SensScal}$ = Sensor Level determined F_{Scal} matrix.

$\underline{\omega}'^{*}$ = $\underline{\omega}'$ based on using a Sensor Level determined F_{Scal} matrix.

$F_{SensAlgn}$ = Pre System Level determined F_{Algn} matrix. Normally, $F_{SensAlgn}$ would be identity (i.e., assumed zero misalignments). For generality, and to make the equations to follow also compatible with coefficient updates following factory calibration, $F_{SensAlgn}$ has been included in Equation (8.1.1.1.1-2).

$\underline{\omega}^{*}$ = $\underline{\omega}$ based on using $\underline{\omega}'^{*}$ (rather than $\underline{\omega}'$).

$\delta\underline{\omega}_{SensBias}$ = Sensor Level determined $\delta\underline{\omega}_{Bias}$ bias vector.

$\delta\underline{\omega}^{*}_{Quant}$ = $\delta\underline{\omega}_{Quant}$ calculated using Sensor Level compensated data.

The $\underline{\omega}_{Puls}$ term in (8.1.1.1.1-1) can be expressed as a function of the true angular rate input vector $\underline{\omega}$ using Equations (8.1.1.1-2) and (8.1.1.1-4) combined:

$$\underline{\omega}_{Puls} = \frac{1}{\Omega_{Wt_0}}\left[\left(I + F_{Scal}\right)\left(F_{Algn}\,\underline{\omega} + \delta\underline{\omega}_{Bias}\right) + \text{Quantization terms}\right] \qquad (8.1.1.1.1\text{-}3)$$

Combining (8.1.1.1.1-1) - (8.1.1.1.1-3) and assuming that quantization effects have been properly compensated in (8.1.1.1.1-2) then yields:

$$\underline{\omega}^* = F_{SensAlgn}^{-1}\left[\left(I + F_{SensScal}\right)^{-1}\left(I + F_{Scal}\right)\left(F_{Algn}\,\underline{\omega} + \delta\underline{\omega}_{Bias}\right) - \delta\underline{\omega}_{SensBias}\right] \qquad (8.1.1.1.1\text{-}4)$$

The System Level sensor calibration procedure is based on measuring the errors induced in strapdown inertial navigation software operations when using the $\underline{\omega}^*$ Sensor Level calibrated strapdown sensor assembly data for input. The System Level calibration procedure (such as described in Chapter 18) typically evaluates angular rate sensor System Level scale factor, misalignment, and bias errors based on the following linearized model for the $\underline{\omega}^*$ input data:

$$\underline{\omega}^* = \left(I + \kappa_{SystScal/Mis}\right)\underline{\omega} + \underline{\kappa}_{SystBias} \qquad (8.1.1.1.1\text{-}5)$$

where

$\kappa_{SystScal/Mis}$ = Angular rate sensor triad System Level Scale-Factor/Misalignment error matrix.

$\underline{\kappa}_{SystBias}$ = Angular rate sensor triad System Level bias error vector.

Equating (8.1.1.1.1-4) and (8.1.1.1.1-5) enables us to solve for new (revised) F_{Scal}, F_{Algn}, $\delta\underline{\omega}_{Bias}$ calibration coefficients in terms of Sensor Level coefficients and the System Level determined values for $\kappa_{SystScal/Mis}$, $\underline{\kappa}_{SystBias}$:

$$\begin{aligned}&\left(I + \kappa_{SystScal/Mis}\right)\underline{\omega} + \underline{\kappa}_{SystBias}\\ &\quad = F_{SensAlgn}^{-1}\left[\left(I + F_{SensScal}\right)^{-1}\left(I + F_{Scal}\right)\left(F_{Algn}\,\underline{\omega} + \delta\underline{\omega}_{Bias}\right) - \delta\underline{\omega}_{SensBias}\right]\end{aligned} \qquad (8.1.1.1.1\text{-}6)$$

Rearranging (8.1.1.1.1-6) obtains:

$$\begin{aligned}&\left(I + F_{Scal}\right)\left(F_{Algn}\,\underline{\omega} + \delta\underline{\omega}_{Bias}\right)\\ &\quad = \left(I + F_{SensScal}\right)\left\{F_{SensAlgn}\left[\left(I + \kappa_{SystScal/Mis}\right)\underline{\omega} + \underline{\kappa}_{SystBias}\right] + \delta\underline{\omega}_{SensBias}\right\}\end{aligned} \qquad (8.1.1.1.1\text{-}7)$$

Equating terms multiplying and not multiplying $\underline{\omega}$ then gives:

$$\left(I + F_{Scal}\right)F_{Algn} = \left(I + F_{SensScal}\right)F_{SensAlgn}\left(I + \kappa_{SystScal/Mis}\right) \qquad (8.1.1.1.1\text{-}8)$$

$$\delta\underline{\omega}_{Bias} = \left(I + F_{Scal}\right)^{-1}\left(I + F_{SensScal}\right)\left[F_{SensAlgn}\,\underline{\kappa}_{SystBias} + \delta\underline{\omega}_{SensBias}\right] \qquad (8.1.1.1.1\text{-}9)$$

Equation (8.1.1.1.1-9) is expanded by first writing from (8.1.1.1.1-8):

$$\left(I + F_{Scal}\right)^{-1}\left(I + F_{SensScal}\right) = F_{Algn}\left[F_{SensAlgn}\left(I + \kappa_{SystScal/Mis}\right)\right]^{-1} \qquad (8.1.1.1.1\text{-}10)$$

Substituting (8.1.1.1.1-10) in (8.1.1.1.1-9) then yields:

$$\begin{aligned}\delta\underline{\omega}_{Bias} &= F_{Algn}\left[F_{SensAlgn}\left(I + \kappa_{SystScal/Mis}\right)\right]^{-1}\left[F_{SensAlgn}\,\underline{\kappa}_{SystBias} + \delta\underline{\omega}_{SensBias}\right] \\ &= F_{Algn}\left(I + \kappa_{SystScal/Mis}\right)^{-1} F_{SensAlgn}^{-1}\left[F_{SensAlgn}\,\underline{\kappa}_{SystBias} + \delta\underline{\omega}_{SensBias}\right] \qquad (8.1.1.1.1\text{-}11) \\ &= F_{Algn}\left(I + \kappa_{SystScal/Mis}\right)^{-1}\left[\underline{\kappa}_{SystBias} + F_{SensAlgn}^{-1}\,\delta\underline{\omega}_{SensBias}\right]\end{aligned}$$

Applying (8.1.1.1.1-11) in (8.1.1.1-7) also obtains for $\underline{K}_{Bias}$ in alternate compensation Equations (8.1.1.1-8):

$$\underline{K}_{Bias} = \left(I + \kappa_{SystScal/Mis}\right)^{-1}\left[\underline{\kappa}_{SystBias} + F_{SensAlgn}^{-1}\,\delta\underline{\omega}_{SensBias}\right] \qquad (8.1.1.1.1\text{-}12)$$

Returning to Equation (8.1.1.1.1-8), we can solve individually for F_{Scal} and F_{Algn} by noting that F_{Scal} is a diagonal matrix, hence:

$$F_{Scal} = \mathrm{Diag}\left[\left(I + F_{SensScal}\right) F_{SensAlgn}\left(I + \kappa_{SystScal/Mis}\right) F_{Algn}^{-1} - I\right] \qquad (8.1.1.1.1\text{-}13)$$

$$F_{Algn_{Off\text{-}Diag}} = \mathrm{Off\ Diag}\left[\left(I + F_{Scal}\right)^{-1}\left(I + F_{SensScal}\right) F_{SensAlgn}\left(I + \kappa_{SystScal/Mis}\right)\right] \qquad (8.1.1.1.1\text{-}14)$$

where

$F_{Algn\,Off\text{-}Diag}$ = Off-diagonal elements of F_{Algn}.

Off Diag ( ) = Off-diagonal elements of ( ).

Diag ( ) = Diagonal elements of ( ).

Since F_{Scal} includes non-linear as well as linear scale factor effects, the non-linear terms (albeit small compared to the linear terms) make F_{Scal} a function of angular rate. Section 8.1.1.3 discusses how Equations (8.1.1.1.1-9) - (8.1.1.1.1-14) (and subsequent developments thereof) can be extended to account for the non-linear effects.

The diagonal elements of F_{Algn} are calculated from the constraint that the rows of F_{Algn} represent unit vectors along sensor input axes. We define:

$$\begin{bmatrix} F_{Algn\,XX} & F_{Algn\,XY} & F_{Algn\,XZ} \\ F_{Algn\,YX} & F_{Algn\,YY} & F_{Algn\,YZ} \\ F_{Algn\,ZX} & F_{Algn\,ZY} & F_{Algn\,ZZ} \end{bmatrix} \equiv F_{Algn} \qquad (8.1.1.1.1\text{-}15)$$

Applying the unit vector constraint to each row of F_{Algn} sets the sum of the squares of each row to unity, hence, the diagonal elements of F_{Algn} are:

$$\begin{aligned} F_{Algn\,XX} &= \sqrt{1 - F^2_{Algn\,XY} - F^2_{Algn\,XZ}} \\ F_{Algn\,YY} &= \sqrt{1 - F^2_{Algn\,YX} - F^2_{Algn\,YZ}} \\ F_{Algn\,ZZ} &= \sqrt{1 - F^2_{Algn\,ZX} - F^2_{Algn\,ZY}} \end{aligned} \tag{8.1.1.1.1-16}$$

Equations (8.1.1.1.1-13) - (8.1.1.1.1-16) must be solved simultaneously to obtain F_{Scal} and F_{Algn}. However, because these equations are non-linear in F_{Scal} and F_{Algn}, a closed-form solution is not possible, and we must resort to iterative numerical techniques. The method is simply to first approximate F_{Algn} in (8.1.1.1.1-13) as $F_{SensAlgn}$, and then solve Equations (8.1.1.1.1-13) - (8.1.1.1.1-16) in the sequence shown to determine F_{Scal} and F_{Algn}. Subsequent iteration cycles use the F_{Algn} value determined from the last iteration cycle in (8.1.1.1.1-13). After two or three iteration cycles, F_{Scal} and F_{Algn} will be calculated to high precision. Equations (8.1.1.1-5) and (8.1.1.1-6) can then be used to obtain Ω_{Wt} and K_{Mis} in alternate compensation Equations (8.1.1.1-8).

It is instructive to apply linearization techniques to Equations (8.1.1.1.1-13) and (8.1.1.1.1-14) to verify that the resulting first order forms are reasonable. We first write:

$$\begin{aligned} F_{Algn} &= I + \left(F_{Algn} - I\right) \\ F_{SensAlgn} &= I + \left(F_{SensAlgn} - I\right) \end{aligned} \tag{8.1.1.1.1-17}$$

The $\left(F_{Algn} - I\right)$ and $\left(F_{SensAlgn} - I\right)$ terms in (8.1.1.1.1-17) are small because F_{Algn} and $F_{SensAlgn}$ are approximately identity. Additionally, from the definition of the rows of F_{Algn} and $F_{SensAlgn}$ as unit vectors, we can also state as in (8.1.1.1.1-16), that the diagonal elements in F_{Algn} and $F_{SensAlgn}$ can be approximated by unity to first order accuracy (first order in the off-diagonal F_{Algn}, $F_{SensAlgn}$ terms). Using these approximations, we now substitute (8.1.1.1.1-17) into Equations (8.1.1.1.1-13) and (8.1.1.1.1-14) while also recognizing that $F_{SensScal}$, F_{Scal} and $\kappa_{SystScal/Mis}$ are small compared to the identity matrix. Expanding the results, dropping products of $F_{SensScal}$, F_{Scal}, $\kappa_{SystScal/Mis}$, $\left(F_{Algn} - I\right)$, $\left(F_{SensAlgn} - I\right)$ as second order (and higher), and approximating the F_{Algn} diagonal elements by unity yields the following first order accuracy version of Equations (8.1.1.1.1-13), (8.1.1.1.1-14) and (8.1.1.1.1-16):

$$F_{Scal} \approx F_{SensScal} + \mathrm{Diag}\left(\kappa_{SystScal/Mis}\right) \tag{8.1.1.1.1-18}$$

$$F_{Algn} \approx F_{SensAlgn} + \mathrm{Off\ Diag}\left(\kappa_{SystScal/Mis}\right) \tag{8.1.1.1.1-19}$$

Equations (8.1.1.1.1-18) and (8.1.1.1.1-19) have the form that one would logically expect (i.e., they make sense).

In practice, linearized forms of Equations (8.1.1.1.1-13) - (8.1.1.1.1-14) can be utilized in many applications based on the assumption that $\kappa_{SystScal/Mis}$, $(F_{Algn} - I)$ and $(F_{SensAlgn} - I)$ will be small while allowing for the possibility that $F_{SensScal}$ may be larger than acceptable for first order approximation. Following the linearization procedure leading to Equations (8.1.1.1.1-18) and (8.1.1.1.1-19), the simplified linearized form of Equation (8.1.1.1.1-13) is obtained from:

$$\begin{aligned} F_{Scal} &= \mathrm{Diag}\left\{(I + F_{SensScal})\left[I + (F_{SensAlgn} - I)\right](I + \kappa_{SystScal/Mis})\left[I + (F_{Algn} - I)\right]^{-1} - I\right\} \\ &\approx \mathrm{Diag}\left\{(I + F_{SensScal})\left[I + (F_{SensAlgn} - I) + \kappa_{SystScal/Mis} - (F_{Algn} - I)\right] - I\right\} \\ &= \mathrm{Diag}\left\{(I + F_{SensScal})(I + F_{SensAlgn} - F_{Algn} + \kappa_{SystScal/Mis}) - I\right\} \\ &= \mathrm{Diag}\left\{F_{SensScal} + (I + F_{SensScal})(F_{SensAlgn} - F_{Algn} + \kappa_{SystScal/Mis})\right\} \end{aligned} \quad (8.1.1.1.1\text{-}20)$$

Equation (8.1.1.1.1-20) can be further reduced using the following general matrix property (which is easily verified by component expansion), that if D is a diagonal matrix and E is a general arbitrary matrix, then:

$$\mathrm{Diag}\,(D\,E) = D\,\mathrm{Diag}\,(E) \quad (8.1.1.1.1\text{-}21)$$

where

D = Arbitrary diagonal matrix.

E = Arbitrary matrix in general.

Since $F_{SensScal}$ is diagonal and (from (8.1.1.1.1-16)) the diagonal elements of $F_{SensAlgn}$ and F_{Algn} are identity to first order, (8.1.1.1.1-20) using (8.1.1.1.1-21) becomes the simplified form:

$$F_{Scal} \approx F_{SensScal} + (I + F_{SensScal})\,\mathrm{Diag}\,(\kappa_{SystScal/Mis}) \quad (8.1.1.1.1\text{-}22)$$

The corresponding linearized form of (8.1.1.1.1-14) is obtained by first noting from (8.1.1.1.1-22) that:

$$I + F_{Scal} \approx (I + F_{SensScal})\left[I + \mathrm{Diag}\,(\kappa_{SystScal/Mis})\right] \quad (8.1.1.1.1\text{-}23)$$

or

$$(I + F_{Scal})^{-1} \approx \left[I + \mathrm{Diag}\,(\kappa_{SystScal/Mis})\right]^{-1}(I + F_{SensScal})^{-1} \quad (8.1.1.1.1\text{-}24)$$

Substituting (8.1.1.1.1-24) into (8.1.1.1.1-14) shows that:

$$
\begin{aligned}
F_{Algn\,Off\text{-}Diag} &\approx \text{Off Diag}\left\{\left[I + \text{Diag}\left(\kappa_{SystScal/Mis}\right)\right]^{-1}\left[I + \left(F_{SensAlgn} - I\right)\right]\left(I + \kappa_{SystScal/Mis}\right)\right\} \\
&\approx \text{Off Diag}\left[I - \text{Diag}\left(\kappa_{SystScal/Mis}\right) + \left(F_{SensAlgn} - I\right) + \kappa_{SystScal/Mis}\right] \qquad (8.1.1.1.1\text{-}25) \\
&= \text{Off Diag}\left(F_{SensAlgn} + \kappa_{SystScal/Mis}\right)
\end{aligned}
$$

Then, because the diagonal elements of F_{Algn} and $F_{SensAlgn}$ are identity (to first order), we have from (8.1.1.1.1-25):

$$
\begin{aligned}
F_{Algn} &\approx I + \text{Off Diag}\left(F_{Algn}\right) \\
&\approx I + \text{Off Diag}\left(F_{SensAlgn}\right) + \text{Off Diag}\left(\kappa_{SystScal/Mis}\right)
\end{aligned}
\qquad (8.1.1.1.1\text{-}26)
$$

or

$$
F_{Algn} \approx F_{SensAlgn} + \text{Off Diag}\left(\kappa_{SystScal/Mis}\right) \qquad (8.1.1.1.1\text{-}27)
$$

Finally, we note that if simplified Equations (8.1.1.1.1-22) and (8.1.1.1.1-27) are to be applied for F_{Scal} and F_{Algn}, a corresponding approximate form of (8.1.1.1.1-12) for $\underline{K}_{Bias}$ should also be used based on the same assumptions that led to the approximate F_{Scal} and F_{Algn} forms; i.e., that $\kappa_{SystScal/Mis}$, $\left(F_{Algn} - I\right)$ and $\left(F_{SensAlgn} - I\right)$ will be small compared to identity. Incorporating these approximations in (8.1.1.1.1-12) then finds:

$$
\underline{K}_{Bias} \approx \underline{\kappa}_{SystBias} + \delta\underline{\omega}_{SensBias} \qquad (8.1.1.1.1\text{-}28)
$$

8.1.1.2 ACCELEROMETER ERROR CHARACTERISTICS AND COMPENSATION FORMULAS

The output vector from a strapdown accelerometer triad can be characterized as a function of its input specific force acceleration vector as:

$$
\begin{aligned}
\underline{a}_{SF_{Puls}} = \frac{1}{A_{Wt_0}}\left(I + G_{Scal}\right)\Big(&G_{Algn}\,\underline{a}_{SF} + \delta\underline{a}_{Bias} \\
&+ \delta\underline{a}_{Size} + \delta\underline{a}_{Aniso} + \delta\underline{a}_{Quant} + \delta\underline{a}_{Rand}\Big)
\end{aligned}
\qquad (8.1.2\text{-}1)
$$

where

$\underline{a}_{SF}$ = Specific force acceleration vector sensed by the accelerometer triad. The components of $\underline{a}_{SF}$ are the specific force acceleration vector projections onto the sensor (or "Body") B Frame.

$\underline{a}_{SF_{Puls}}$ = Accelerometer triad output vector in pulses per second. Each axis output pulse is a digital indication that the sensor associated with that axis has accelerated through an integrated specific force acceleration increment along its input axis equal to that particular sensor's pulse size.

Aw_{t_0} = Nominal pulse weight (a positive value) for each accelerometer (ft per sec per pulse).

I = Identity matrix.

G_{Scal} = Accelerometer triad scale factor correction matrix; a diagonal matrix in which each element adjusts the output pulse scaling to correspond to the actual scaling for the particular sensor output. Nominally, the G_{Scal} matrix is zero. The G_{Scal} matrix may include non-linear scale factor effects and temperature dependency.

G_{Algn} = Alignment matrix for the accelerometer triad. Each row represents a unit vector along a particular accelerometer input axis as projected onto the B-Frame. Nominally, the G_{Algn} matrix is identity. The G_{Algn} matrix may include specific force acceleration dependency.

$\delta\underline{a}_{Bias}$ = Accelerometer triad bias vector. Each element equals the systematic output from a particular accelerometer under zero input specific force acceleration conditions. In some accelerometers, $\delta\underline{a}_{Bias}$ may have environmental sensitivities (e.g., temperature and linear vibration dependency).

$\delta\underline{a}_{Size}$ = Accelerometer triad size effect error created by the fact that due to physical size, the accelerometers in the triad cannot be collocated, hence, do not measure components of identically the same acceleration vector (See Section 8.1.4.1).

$\delta\underline{a}_{Aniso}$ = Accelerometer triad anisoinertia error effect (present in pendulous accelerometers) created by mismatch in the moments of inertia around the input and pendulum axes (See Section 8.1.4.2).

$\delta\underline{a}_{Quant}$ = Instantaneous accelerometer triad pulse quantization error associated with the output only being provided when the cumulative input equals the pulse weight per axis. Includes pulse output logic dead-band effect under turn-around conditions (See Section 8.1.3.2).

$\delta\underline{a}_{Rand}$ = Accelerometer triad random error output vector.

As in Section 8.1.1.1, Equation (8.1.1.2-1) can be solved for the B Frame specific force acceleration input vector as follows:

$$\underline{a}'_{SF} = Aw_{t_0}\left(I + G_{Scal}\right)^{-1} \underline{a}_{SF_{Puls}} \qquad (8.1.1.2\text{-}2)$$

$$\underline{a}_{SF} = G_{Algn}^{-1}\left(\underline{a}'_{SF} - \delta\underline{a}_{Bias} - \delta\underline{a}_{Size} - \delta\underline{a}_{Aniso} - \delta\underline{a}_{Quant} - \delta\underline{a}_{Rand}\right) \qquad (8.1.1.2\text{-}3)$$

where

$\underline{a}_{SF}^{\prime}$ = Scale factor compensated accelerometer triad output vector.

Equation (8.1.1.2-2) represents the scale factor compensation equation for the raw accelerometer triad $\underline{a}_{SF_{Puls}}$ output. Note that because G_{Scal} is a diagonal matrix, the inverse of $\left(I + G_{Scal}\right)$ in Equation (8.1.1.2-2) simply equals the matrix with diagonal elements equal to the reciprocal of the $\left(I + G_{Scal}\right)$ diagonal terms. Compensation for the remaining predictable errors in $\underline{a}_{SF_{Puls}}$ is achieved using a simplified form of (8.1.1.2-3) in which it is recognized that the $\delta\underline{a}_{Rand}$ component is unpredictable, hence, can only be approximated by zero:

$$\underline{a}_{SF} = G_{Algn}^{-1}\left(\underline{a}_{SF}^{\prime} - \delta\underline{a}_{Bias} - \delta\underline{a}_{Size} - \delta\underline{a}_{Aniso} - \delta\underline{a}_{Quant}\right) \tag{8.1.1.2-4}$$

Compensation Equations (8.1.1.2-2) and (8.1.1.2-4) are further refined to a more familiar form by introducing the following definitions:

$$A_{Wt} \equiv A_{Wt_0}\left(I + G_{Scal}\right)^{-1} \tag{8.1.1.2-5}$$

$$L_{Mis} \equiv I - G_{Algn}^{-1} \tag{8.1.1.2-6}$$

$$\underline{L}_{Bias} \equiv G_{Algn}^{-1}\,\delta\underline{a}_{Bias} \tag{8.1.1.2-7}$$

where

A_{Wt} = Accelerometer triad scale factor pulse weighting matrix (fps per pulse).

L_{Mis} = Accelerometer triad misalignment compensation matrix (nominally zero).

$\underline{L}_{Bias}$ = Accelerometer bias compensation vector.

Substituting (8.1.1.2-5) - (8.1.1.2-7) into (8.1.1.2-2) and (8.1.1.2-4) then obtains the equivalent compensation equations:

$$\begin{aligned} \underline{a}_{SF}^{\prime} &= A_{Wt}\,\underline{a}_{SF_{Puls}} \\ \underline{a}_{SF} &\approx \underline{a}_{SF}^{\prime} - L_{Mis}\,\underline{a}_{SF}^{\prime} - \underline{L}_{Bias} - G_{Algn}^{-1}\left(\delta\underline{a}_{Size} + \delta\underline{a}_{Aniso} + \delta\underline{a}_{Quant}\right) \end{aligned} \tag{8.1.1.2-8}$$

Equations (8.1.1.2-5) - (8.1.1.2-8) constitute the compensation equations for the accelerometer triad output vector in continuous acceleration vector format. These equations will form the basis for the accelerometer triad compensation algorithms presented in Section 8.1.2.2.

8.1.1.2.1 Sensor And System Level Compensation Coefficient Evaluation For The Accelerometers

Compensation of the $\underline{a}_{SF_{Puls}}$ accelerometer triad as described in Section 8.1.1.2 requires that G_{Scal}, G_{Algn} and $\delta\underline{a}_{Bias}$ be evaluated for the accelerometers. This typically involves use of generic analytic algorithms that characterize the sensor performance in terms of coefficients that are unique to each individual sensor. Determination of the coefficients is the calibration process performed as part of manufacturing/assembly/test operations. A typical "factory" calibration process consists of two steps: “Sensor Level” calibration and “System Level” calibration. Sensor Level calibration is performed on each individual sensor before installation into the strapdown sensor assembly, and typically determines (for the accelerometers) the primary contributors to G_{Scal} and $\delta\underline{a}_{Bias}$. Following Sensor Level calibration, the sensors are installed in the sensor assembly and System Level calibration is performed. System Level calibration then determines (for the accelerometers) the G_{Algn} matrix and residual corrections to G_{Scal} and $\delta\underline{a}_{Bias}$.

System Level calibration is performed on a sensor assembly that is being compensated using Sensor Level compensation equations, hence, in effect, the following forms of Equations (8.1.1.1-2) and (8.1.1.1-4) apply for the accelerometer outputs:

$$\underline{a}'_{SF}{}^{*} = A_{Wt_0}\left(I + G_{SensScal}\right)^{-1} \underline{a}_{SF_{Puls}} \tag{8.1.1.2.1-1}$$

$$\underline{a}^{*}_{SF} = G^{-1}_{SensAlgn}\left(\underline{a}'_{SF}{}^{*} - \delta\underline{a}_{SensBias} - \delta\underline{a}^{*}_{Size} - \delta\underline{a}^{*}_{Aniso} - \delta\underline{a}^{*}_{Quant}\right) \tag{8.1.1.2.1-2}$$

where

$G_{SensScal}$ = Sensor Level determined G_{Scal} matrix.

$\underline{a}'_{SF}{}^{*}$ = $\underline{a}'_{SF}$ based on using a Sensor Level determined G_{Scal} matrix.

$G_{SensAlgn}$ = Pre System Level determined G_{Algn} matrix. Normally, $G_{SensAlgn}$ would be identity (i.e., assumed zero misalignment). For generality, and to make the equations to follow also compatible with coefficient updates following factory calibration, $G_{SensAlgn}$ has been included in Equation (8.1.1.2.1-2).

$\underline{a}^{*}_{SF}$ = $\underline{a}_{SF}$ based on using $\underline{a}'_{SF}{}^{*}$ (rather than $\underline{a}'_{SF}$).

$\delta\underline{a}_{SensBias}$ = Sensor Level determined $\delta\underline{a}_{Bias}$ bias vector.

$\delta\underline{a}^{*}_{Size}$, $\delta\underline{a}^{*}_{Aniso}$, $\delta\underline{a}^{*}_{Quant}$ = $\delta\underline{a}_{Size}$, $\delta\underline{a}_{Aniso}$, $\delta\underline{a}_{Quant}$ calculated using Sensor Level compensated data.

The $\underline{a}_{SF_{Puls}}$ term in (8.1.1.2.1-1) can be expressed as a function of the true specific force acceleration input vector $\underline{a}_{SF}$ using Equations (8.1.1.2-2) and (8.1.1.2-4) combined:

$$\underline{a}_{SF_{Puls}} = \frac{1}{A_{Wt_0}} \left[\left(I + G_{Scal}\right)\left(G_{Algn}\, \underline{a}_{SF} + \delta\underline{a}_{Bias}\right) + \begin{matrix}\text{Size Effect, Anisoinertia,} \\ \text{and Quantization terms}\end{matrix} \right] \qquad (8.1.1.2.1\text{-}3)$$

Combining (8.1.1.2.1-1) - (8.1.1.2.1-3) and assuming that size effect, anisoinertia, and quantization effects have been properly compensated in (8.1.1.2.1-2) then yields:

$$\underline{a}^{*}_{SF} = G^{-1}_{SensAlgn} \left[\left(I + G_{SensScal}\right)^{-1} \left(I + G_{Scal}\right)\left(G_{Algn}\, \underline{a}_{SF} + \delta\underline{a}_{Bias}\right) - \delta\underline{a}_{SensBias} \right] \qquad (8.1.1.2.1\text{-}4)$$

The System Level sensor calibration procedure is based on measuring the errors induced in strapdown inertial navigation software operations when using the $\underline{a}^{*}_{SF}$ Sensor Level calibrated strapdown sensor assembly data for input. The System Level calibration procedure (such as described in Chapter 18) typically evaluates accelerometer System Level scale factor, misalignment, and bias errors based on the following linearized model for the $\underline{a}^{*}_{SF}$ input data:

$$\underline{a}^{*}_{SF} = \left(I + \lambda_{SystScal/Mis}\right) \underline{a}_{SF} + \underline{\lambda}_{SystBias} \qquad (8.1.1.2.1\text{-}5)$$

where

$\lambda_{SystScal/Mis}$ = Accelerometer triad System Level scale-factor/misalignment error matrix.

$\underline{\lambda}_{SystBias}$ = Accelerometer triad System Level bias error vector.

Equating (8.1.1.2.1-4) and (8.1.1.2.1-5) enables us to solve for new (revised) G_{Scal}, G_{Algn}, $\delta\underline{a}_{Bias}$ calibration coefficients in terms of Sensor Level coefficients and the System Level determined values for $\lambda_{SystScal/Mis}$, $\underline{\lambda}_{SystBias}$. Following the same procedure leading to Equations (8.1.1.1.1-8) and (8.1.1.1.1-9), we find for the accelerometers:

$$\left(I + G_{Scal}\right) G_{Algn} = \left(I + G_{SensScal}\right) G_{SensAlgn} \left(I + \lambda_{SystScal/Mis}\right) \qquad (8.1.1.2.1\text{-}6)$$

$$\delta\underline{a}_{Bias} = \left(I + G_{Scal}\right)^{-1} \left(I + G_{SensScal}\right) \left[G_{SensAlgn}\, \underline{\lambda}_{SystBias} + \delta\underline{a}_{SensBias} \right] \qquad (8.1.1.2.1\text{-}7)$$

Equation (8.1.1.2.1-7) can be used to solve for $\underline{L}_{Bias}$ in (8.1.1.2-7) by first writing from (8.1.1.2.1-6):

$$\left(I + G_{Scal}\right)^{-1} \left(I + G_{SensScal}\right) = G_{Algn} \left[G_{SensAlgn} \left(I + \lambda_{SystScal/Mis}\right) \right]^{-1} \qquad (8.1.1.2.1\text{-}8)$$

Substituting (8.1.1.2.1-8) in (8.1.1.2.1-7) then yields for $\delta\underline{a}_{Bias}$:

$$
\begin{aligned}
\delta\underline{a}_{Bias} &= G_{Algn}\left[G_{SensAlgn}\left(I + \lambda_{SystScal/Mis}\right)\right]^{-1}\left[G_{SensAlgn}\,\underline{\lambda}_{SystBias} + \delta\underline{a}_{SensBias}\right] \\
&= G_{Algn}\left(I + \lambda_{SystScal/Mis}\right)^{-1} G_{SensAlgn}^{-1}\left[G_{SensAlgn}\,\underline{\lambda}_{SystBias} + \delta\underline{a}_{SensBias}\right] \qquad (8.1.1.2.1\text{-}9) \\
&= G_{Algn}\left(I + \lambda_{SystScal/Mis}\right)^{-1}\left[\underline{\lambda}_{SystBias} + G_{SensAlgn}^{-1}\,\delta\underline{a}_{SensBias}\right]
\end{aligned}
$$

Applying (8.1.1.2.1-9) in (8.1.1.2-7) also obtains for $\underline{L}_{Bias}$:

$$
\underline{L}_{Bias} = \left(I + \lambda_{SystScal/Mis}\right)^{-1}\left[\underline{\lambda}_{SystBias} + G_{SensAlgn}^{-1}\,\delta\underline{a}_{SensBias}\right] \qquad (8.1.1.2.1\text{-}10)
$$

Returning to Equation (8.1.1.2.1-6) and following the same procedure leading to Equations (8.1.1.1.1-13) - (8.1.1.1.1-16), we obtain the iterative form of Equation (8.1.1.2.1-6) to be used in evaluating G_{Scal}, G_{Algn} for the accelerometers:

$$
G_{Scal} = \mathrm{Diag}\left[\left(I + G_{SensScal}\right) G_{SensAlgn}\left(I + \lambda_{SystScal/Mis}\right) G_{Algn}^{-1} - I\right] \qquad (8.1.1.2.1\text{-}11)
$$

$$
G_{Algn_{Off\text{-}Diag}} = \mathrm{Off\ Diag}\left[\left(I + G_{Scal}\right)^{-1}\left(I + G_{SensScal}\right) G_{SensAlgn}\left(I + \lambda_{SystScal/Mis}\right)\right] \qquad (8.1.1.2.1\text{-}12)
$$

$$
\begin{bmatrix} G_{Algn\,XX} & G_{Algn\,XY} & G_{Algn\,XZ} \\ G_{Algn\,YX} & G_{Algn\,YY} & G_{Algn\,YZ} \\ G_{Algn\,ZX} & G_{Algn\,ZY} & G_{Algn\,ZZ} \end{bmatrix} \equiv G_{Algn} \qquad (8.1.1.2.1\text{-}13)
$$

$$
\begin{aligned}
G_{Algn\,XX} &= \sqrt{1 - G_{Algn\,XY}^2 - G_{Algn\,XZ}^2} \\
G_{Algn\,YY} &= \sqrt{1 - G_{Algn\,YX}^2 - G_{Algn\,YZ}^2} \qquad (8.1.1.2.1\text{-}14) \\
G_{Algn\,ZZ} &= \sqrt{1 - G_{Algn\,ZX}^2 - G_{Algn\,ZY}^2}
\end{aligned}
$$

where

$G_{Algn\,Off\text{-}Diag}$ = Off-diagonal elements of G_{Algn}.

Since G_{Scal} includes non-linear as well as linear scale factor effects, the non-linear terms (albeit generally small compared to the linear terms) make G_{Scal} a function of specific force. Section 8.1.1.3 discusses how Equations (8.1.1.2.1-11) - (8.1.1.2.1-14) (and subsequent developments thereof in this section) can be extended to account for the non-linear effects.

Equations (8.1.1.2.1-11) - (8.1.1.2.1-14) are solved iteratively by first approximating G_{Algn} in Equation (8.1.1.2.1-11) as $G_{SensAlgn}$, and then processing Equations (8.1.1.2.1-11) - (8.1.1.2.1-14) in the sequence shown to determine G_{Scal} and G_{Algn}. Subsequent iteration cycles then use the G_{Algn} value determined from the last iteration cycle in Equation (8.1.1.2.1-11). After two or three iteration cycles, G_{Scal} and G_{Algn} will be calculated to high precision.

In practice, linearized forms of Equations (8.1.1.2.1-11) - (8.1.1.2.1-14) can be utilized in many applications based on the assumption that $\lambda_{SystScal/Mis}$, $\left(G_{Algn} - I\right)$ and $\left(G_{SensAlgn} - I\right)$ will be small. Following the linearization procedure that led to Equations (8.1.1.1.1-22), (8.1.1.1.1-27) and (8.1.1.1.1-28), the simplified approximate forms of Equations (8.1.1.2.1-10) - (8.1.1.2.1-14) are given by:

$$G_{Scal} \approx G_{SensScal} + \left(I + G_{SensScal}\right) \text{Diag}\left(\lambda_{SystScal/Mis}\right) \tag{8.1.1.2.1-15}$$

$$G_{Algn} \approx G_{SensAlgn} + \text{Off Diag}\left(\lambda_{SystScal/Mis}\right) \tag{8.1.1.2.1-16}$$

$$\underline{L}_{Bias} \approx \underline{\lambda}_{SystBias} + \delta\underline{a}_{SensBias} \tag{8.1.1.2.1-17}$$

8.1.1.3 DEALING WITH SCALE FACTOR NON-LINEARITIES

In many applications, the Ω_{Wt} and A_{Wt} scale factor matrices in Sections 8.1.1.1.1 and 8.1.1.2.1 are approximated as constant, representing a linear scale factor error characteristic (i.e., angular-rate-sensor/accelerometer output linearly proportional to angular-rate/specific-force input). If non-linear scale factor effects exist, Ω_{Wt} and A_{Wt} become functions of angular-rate/specific-force. In this section we analyze how the results of Sections 8.1.1.1.1 and 8.1.1.2.1 can be expanded to specifically account for scale factor non-linearities. We will develop the procedure first for the accelerometers and then apply it to the angular rate sensors.

The accelerometer triad pulse weighting matrix A_{Wt} is calculated from G_{Scal} as shown in Equation (8.1.1.2-5). To handle non-linearities in A_{Wt}, let us begin by defining G_{Scal} as being composed of linear and non-linear terms, viz.:

$$I + G_{Scal} = \left(I + G_{ScalLin}\right)\left(I + G_{ScalNonLin}\right) \tag{8.1.1.3-1}$$

where

$G_{ScalLin}$ = Linear scale factor portion of G_{Scal} (independent of specific force input).

$G_{ScalNonLin}$ = Non-linear portion of G_{Scal} dependent on specific force input.

We now address the problem of calculating $G_{ScalLin}$ and $G_{ScalNonLin}$ for (8.1.1.3-1) from Sensor Level and System Level test data measurements. As in (8.1.1.3-1) we first define:

$$\begin{aligned} I + G_{SensScal} &= \left(I + G_{SensScalLin}\right)\left(I + G_{SensScalNonLin}\right) \\ \lambda_{SystScal/Mis} &= \lambda_{SystScalLin/Mis} + \lambda_{SystScalNonLin} \end{aligned} \qquad (8.1.1.3\text{-}2)$$

where

$G_{SensScalLin}$ = Linear scale factor portion of $G_{SensScal}$ (independent of specific force input).

$G_{SensScalNonLin}$ = Non-linear portion of $G_{SensScal}$ dependent on specific force input.

$\lambda_{SystScalLin/Mis}$ = Linear scale factor portion of $\lambda_{SystScal/Mis}$ (independent of specific force input).

$\lambda_{SystScalNonLin}$ = Non-linear portion of $\lambda_{SystScal/Mis}$ dependent on specific force input. By its name, we are assuming that the non-linear terms in $\lambda_{SystScalNonLin}$ are only in the scale factor (diagonal elements) and do not include misalignment effects (off-diagonal elements); i.e., $\lambda_{SystScalNonLin}$ is a diagonal matrix.

In general, $G_{SensScalNonLin}$ and $G_{SensScalLin}$ would be determined from Sensor Level tests with $G_{SensScalNonLin}$ calculated as a function of the raw accelerometer pulse output (see Equation (8.1.1.2-1) for general accelerometer output model), but applied during Sensor Level testing on each individual accelerometer. The $\lambda_{SystScalLin/Mis}$ and $\lambda_{SystScalNonLin}$ terms would be determined by System Level tests, typically based on a linearized error model such as Equation (8.1.1.2.1-5). Chapter 18, Section 18.4 provides an example of a typical System Level test and associated error determination procedure.

Continuing, from Equation (8.1.1.2.1-15) we see that $\left(I + G_{Scal}\right)$ in (8.1.1.3-1) is also given by:

$$I + G_{Scal} \approx \left(I + G_{SensScal}\right)\left[I + \mathrm{Diag}\left(\lambda_{SystScal/Mis}\right)\right] \qquad (8.1.1.3\text{-}3)$$

Substituting (8.1.1.3-1) and (8.1.1.3-2) into (8.1.1.3-3) obtains:

$$\begin{aligned} \left(I + G_{ScalLin}\right)\left(I + G_{ScalNonLin}\right) &= \left(I + G_{SensScalLin}\right)\left(I + G_{SensScalNonLin}\right)\left[I \right. \\ &\quad \left. + \mathrm{Diag}\left(\lambda_{SystScalLin/Mis}\right) + \lambda_{SystScalNonLin}\right] \qquad (8.1.1.3\text{-}4) \\ &\approx \left(I + G_{SensScalLin}\right)\left(I + G_{SensScalNonLin}\right)\left[I + \mathrm{Diag}\left(\lambda_{SystScalLin/Mis}\right)\right]\left(I + \lambda_{SystScalNonLin}\right) \end{aligned}$$

Recognizing that all matrices in (8.1.1.3-4) are diagonal and that the product of diagonal matrices is unaffected by the order of multiplication, Equation (8.1.1.3-4) can be rearranged into the following:

$$\begin{aligned} &\left(I + G_{ScalLin}\right)\left(I + G_{ScalNonLin}\right) \\ &\approx \left(I + G_{SensScalLin}\right)\left[I + \mathrm{Diag}\left(\lambda_{SystScalLin/Mis}\right)\right]\left(I + G_{SensScalNonLin}\right)\left(I + \lambda_{SystScalNonLin}\right) \end{aligned} \tag{8.1.1.3-5}$$

from which we can then write:

$$\begin{aligned} I + G_{ScalLin} &\approx \left(I + G_{SensScalLin}\right)\left[I + \mathrm{Diag}\left(\lambda_{SystScalLin/Mis}\right)\right] \\ I + G_{ScalNonLin} &\approx \left(I + G_{SensScalNonLin}\right)\left(I + \lambda_{SystScalNonLin}\right) \end{aligned} \tag{8.1.1.3-6}$$

Accelerometer non-linear scale factor effects can be handled using (8.1.1.3-1) for $\left(I + G_{Scal}\right)$ in A_{W_t} Equation (8.1.1.2-5), with $\left(I + G_{ScalLin}\right)$ and $\left(I + G_{ScalNonLin}\right)$ provided by (8.1.1.3-6).

Equivalent results are obtained for the angular rate sensors by applying the same methodology to Equation (8.1.1.1.1-22). The results are:

$$\begin{aligned} I + F_{Scal} &= \left(I + F_{ScalLin}\right)\left(I + F_{ScalNonLin}\right) \\ \kappa_{SystScal/Mis} &= \kappa_{SystScalLin/Mis} + \kappa_{SystScalNonLin} \end{aligned} \tag{8.1.1.3-7}$$

with

$$\begin{aligned} I + F_{ScalLin} &\approx \left(I + F_{SensScalLin}\right)\left[I + \mathrm{Diag}\left(\kappa_{SystScalLin/Mis}\right)\right] \\ I + F_{ScalNonLin} &\approx \left(I + F_{SensScalNonLin}\right)\left(I + \kappa_{SystScalNonLin}\right) \end{aligned} \tag{8.1.1.3-8}$$

where

$F_{ScalLin}$ = Linear scale factor portion of F_{Scal} (independent of angular rate input).

$F_{ScalNonLin}$ = Non-linear portion of F_{Scal} dependent on angular rate input.

$F_{SensScalLin}$ = Linear scale factor portion of $F_{SensScal}$ (independent of angular rate input).

$F_{SensScalNonLin}$ = Non-linear portion of $F_{SensScal}$ dependent on angular rate input.

$\kappa_{SystScalLin/Mis}$ = Linear scale factor portion of $\kappa_{SystScal/Mis}$ (independent of angular rate input).

$\kappa_{SystScalNonLin}$ = Non-linear portion of $\kappa_{SystScal/Mis}$ dependent on angular rate input (a diagonal matrix attributable to only scale factor error effects).

Angular rate sensor non-linear scale factor effects can be handled using (8.1.1.3-7) for $\left(I + F_{Scal}\right)$ in Ω_{Wt} Equation (8.1.1.1-5), with $\left(I + F_{ScalLin}\right)$ and $\left(I + F_{ScalNonLin}\right)$ provided by (8.1.1.3-8). In general, $F_{SensScalNonLin}$ and $F_{SensScalLin}$ would be determined from Sensor Level tests with $F_{SensScalNonLin}$ calculated as a function of the raw angular rate sensor pulse output (see Equation (8.1.1.1-1) for general angular rate sensor output model), but applied during Sensor Level testing on each individual angular rate sensor. The $\kappa_{SystScalLin/Mis}$ and $\kappa_{SystScalNonLin}$ terms would be determined by System Level tests, typically based on a linearized error model such as Equation (8.1.1.1.1-5). Chapter 18, Section 18.4 provides an example of a typical System Level test and associated error determination procedure.

As an example of the application of the above results, let us consider the case of a common accelerometer scale factor asymmetry type non-linearity. Scale factor asymmetry produces an acceleration error proportional to the magnitude of the input acceleration. From Equation (8.1.1.2-1) we see that $\left(I + G_{Scal}\right)$ (a diagonal matrix containing the non-linearity) multiplies a composite of acceleration terms which we define for discussion purposes as $\underline{a}''_{SF}$:

$$\underline{a}''_{SF} \equiv G_{Algn}\, \underline{a}_{SF} + \delta\underline{a}_{Bias} + \delta\underline{a}_{Size} + \delta\underline{a}_{Aniso} + \delta\underline{a}_{Quant} + \delta\underline{a}_{Rand} \qquad (8.1.1.3\text{-}9)$$

To account for asymmetrical scale factor error, $\left(I + G_{Scal}\right)$ should include a constant component ($G_{ScalNonLin}$) that reverses its sign for negative compared to positive values of $\underline{a}''_{SF}$. The desired result is achieved from (8.1.1.2-1) by modeling $\left(I + G_{Scal}\right)$ as in (8.1.1.3-1) with $G_{ScalNonLin}$ given by:

$$G_{ScalNonLin} = G_{ScalAsym}\, A''_{Sign} \qquad (8.1.1.3\text{-}10)$$

in which from (8.1.1.2-1) and (8.1.1.3-9):

$$A''_{Sign} \equiv \begin{bmatrix} \mathrm{Sign}\left(a''_{SF_X}\right) & 0 & 0 \\ 0 & \mathrm{Sign}\left(a''_{SF_Y}\right) & 0 \\ 0 & 0 & \mathrm{Sign}\left(a''_{SF_Z}\right) \end{bmatrix} \qquad (8.1.1.3\text{-}11)$$

where

a''_{SF_i} = Component i of $\underline{a}''_{SF}$ in (8.1.1.3-9).

A''_{Sign} = Diagonal matrix with element i equal to the sign of a''_{SF_i}.

Sign ( ) = One for ( ) greater than or equal to zero and minus one for ( ) less than zero.

$G_{ScalAsym}$ = Accelerometer scale factor asymmetry matrix.

Because G_{Scal} is diagonal and small compared to identity, $\left(I + G_{Scal}\right)$ will be diagonal with positive components. Therefore, from (8.1.1.3-9) and (8.1.1.2-1), the sign of the $\underline{a}_{SF_{Puls}}$ components in (8.1.1.2-1) will equal the sign of the (8.1.1.3-9) $\underline{a}''_{SF}$ components:

$$A''_{Sign} = A_{Puls_{Sign}} \tag{8.1.1.3-12}$$

where

$A_{Puls_{Sign}}$ = Diagonal matrix with element i equal to the sign of a_{SFPuls_i} defined below.

a_{SFPuls_i} = Accelerometer i component of the $\underline{a}_{SF_{Puls}}$ output pulse rate vector.

Thus, using (8.1.1.3-12), Equation (8.1.1.3-10) is equivalently:

$$G_{ScalNonLin} = G_{ScalAsym}\, A_{Puls_{Sign}} \tag{8.1.1.3-13}$$

Similarly, for $G_{SensScalNonLin}$ in (8.1.1.3-2) determined from Sensor Level tests, we write (as in (8.1.1.3-13)) :

$$G_{SensScalNonLin} = G_{SensScalAsym}\, A_{Puls_{Sign}} \tag{8.1.1.3-14}$$

where

$G_{SensScalAsym}$ = Accelerometer scale factor asymmetry matrix determined from Sensor Level testing.

The $\lambda_{SystScalNonLin}$ matrix during System Level tests can be modeled from (8.1.1.2.1-5) and (8.1.1.3-2) as:

$$\lambda_{SystScalNonLin} = \lambda_{SystScalAsym}\, A_{Sign} \tag{8.1.1.3-15}$$

where

A_{Sign} = Diagonal matrix with element i equal to the sign of a_{SF_i} defined below.

a_{SF_i} = Component i of $\underline{a}_{SF}$ in Equation (8.1.1.2.1-5).

$\lambda_{SystScalAsym}$ = Accelerometer scale factor asymmetry matrix determined from System Level testing.

Applying (8.1.1.3-12), we make the first order approximation that:

$$A_{Sign} \approx A''_{Sign} = A_{Puls_{Sign}} \tag{8.1.1.3-16}$$

from which (8.1.1.3-15) becomes:

$$\lambda_{SystScalNonLin} \approx \lambda_{SystScalAsym} \, A_{PulsSign} \tag{8.1.1.3-17}$$

Substituting (8.1.1.3-14) and (8.1.1.3-17) into the (8.1.1.3-6) $\left(I + G_{ScalNonLin}\right)$ expression obtains:

$$I + G_{ScalNonLin} = \left(I + G_{SensScalAsym} A_{Puls_{Sign}}\right)\left(I + \lambda_{SystScalAsym} \, A_{PulsSign}\right) \tag{8.1.1.3-18}$$

Equation (8.1.1.3-18) for $\left(I + G_{ScalNonLin}\right)$ and (8.1.1.3-6) for $\left(I + G_{ScalLin}\right)$, when substituted in (8.1.1.3-1), finds $\left(I + G_{Scal}\right)$ for A_{Wt} in (8.1.1.2-5), which then determines $\underline{a}'_{SF}$ in (8.1.1.2-8). A composite version of these results obtains the following revised form of the $\underline{a}'_{SF}$ expression in (8.1.1.2-8):

$$\underline{a}'_{SF} = A_{Wt_+} \, \underline{a}_{SF+Puls} + A_{Wt_-} \, \underline{a}_{SF-Puls} \tag{8.1.1.3-19}$$

in which

$$\underline{a}_{SF+Puls} \equiv \begin{bmatrix} a_{SF+Puls_X} \\ a_{SF+Puls_Y} \\ a_{SF+Puls_Z} \end{bmatrix} \qquad \underline{a}_{SF-Puls} \equiv \begin{bmatrix} a_{SF-Puls_X} \\ a_{SF-Puls_Y} \\ a_{SF-Puls_Z} \end{bmatrix}$$

$$\begin{aligned} A_{Wt_+} = {} & A_{Wt_0}\Big\{\left(I + G_{SensScalLin}\right)\Big[I \\ & + \mathrm{Diag}\left(\lambda_{SystScalLin/Mis}\right)\Big]\left(I + G_{SensScalAsym}\right)\left(I + \lambda_{SystScalAsym}\right)\Big\}^{-1} \\ A_{Wt_-} = {} & A_{Wt_0}\Big\{\left(I + G_{SensScalLin}\right)\Big[I \\ & + \mathrm{Diag}\left(\lambda_{SystScalLin/Mis}\right)\Big]\left(I - G_{SensScalAsym}\right)\left(I - \lambda_{SystScalAsym}\right)\Big\}^{-1} \end{aligned} \tag{8.1.1.3-20}$$

where

$a_{SF+Puls_i} = a_{SFPuls_i}$ defined previously when it is positive.

$a_{SF-Puls_i} = a_{SFPuls_i}$ when it is negative.

$\underline{a}_{SF+Puls}$ = Accelerometer pulse rate output vector formed from the $a_{SF+Puls_i}$'s.

$\underline{a}_{SF-Puls}$ = Accelerometer pulse rate output vector formed from the $a_{SF-Puls_i}$'s.

A_{Wt_+}, A_{Wt_-} = Accelerometer plus and minus pulse weighting coefficients.

Equation (8.1.1.3-19) lends itself to compensation for a commonly used sensor output format in which positive and negative pulses are provided on separate output lines. Scale factor asymmetry compensation can be achieved in this situation by scaling the positive pulses by

A_{Wt_+}, the negative pulses by A_{Wt_-}, and summing the result to form $\underline{a}'_{SF}$. This is fairly simple and can be achieved on all sensor input data at the maximum input rate, prior to application of remaining compensations. The A_{Wt_+} and A_{Wt_-} matrices would be calculated at system start-up if all contributions are constant, or at a lower rate if some contributions vary with time (e.g., temperature sensitive terms modeled as a function of measured sensor temperature).

During System Level testing, the A_{Wt_+}, A_{Wt_-} values used to compute $\underline{a}'_{SF}$ in accelerometer compensation Equation (8.1.1.3-19) would be as in (8.1.1.3-20), but with the $\lambda_{Syst\ldots}$ matrices set to zero. The results of the System Level testing would, of course, then be used to calculate the $\lambda_{Syst\ldots}$ matrices.

The previous accelerometer scale factor asymmetry results are easily extended to angular rate sensors for the $\underline{\omega}'$ vector in compensation Equations (8.1.1.1-8) which then becomes:

$$\underline{\omega}' = \Omega_{Wt_+}\,\underline{\omega}_{+Puls} + \Omega_{Wt_-}\,\underline{\omega}_{-Puls} \qquad (8.1.1.3\text{-}21)$$

with

$$\underline{\omega}_{+Puls} \equiv \begin{bmatrix} \omega_{+Puls_X} \\ \omega_{+Puls_Y} \\ \omega_{+Puls_Z} \end{bmatrix} \qquad \underline{\omega}_{-Puls} \equiv \begin{bmatrix} \omega_{-Puls_X} \\ \omega_{-Puls_Y} \\ \omega_{-Puls_Z} \end{bmatrix}$$

$$\begin{aligned} \Omega_{Wt_+} = \Omega_{Wt_0}\Big\{&\big(I + F_{SensScalLin}\big)\Big[I \\ &+ \mathrm{Diag}\big(\kappa_{SystScalLin/Mis}\big)\Big]\big(I + F_{SensScalAsym}\big)\big(I + \kappa_{SystScalAsym}\big)\Big\}^{-1} \end{aligned} \qquad (8.1.1.3\text{-}22)$$

$$\begin{aligned} \Omega_{Wt_-} = \Omega_{Wt_0}\Big\{&\big(I + F_{SensScalLin}\big)\Big[I \\ &+ \mathrm{Diag}\big(\kappa_{SystScalLin/Mis}\big)\Big]\big(I - F_{SensScalAsym}\big)\big(I - \kappa_{SystScalAsym}\big)\Big\}^{-1} \end{aligned}$$

where

$F_{SensScalAsym}$ = Angular rate sensor scale factor asymmetry matrix determined from Sensor Level testing.

$\kappa_{SystScalAsym}$ = Angular rate sensor scale factor asymmetry matrix determined from System Level testing.

ω_{+Puls_i} = ω_{Puls_i} defined below when it is positive.

ω_{Puls_i} = Component i of angular rate sensor output pulse rate vector $\underline{\omega}_{Puls}$.

ω_{-Puls_i} = ω_{Puls_i} when it is negative.

$\underline{\omega}_{+Puls}$ = Angular rate sensor pulse rate output vector formed from the ω_{+Puls_i}'s.

$\underline{\omega}_{-Puls}$ = Angular rate sensor pulse rate output vector formed from the ω_{-Puls_i}'s.

Ω_{Wt_+}, Ω_{Wt_-} = Angular rate sensor plus and minus pulse weighting coefficients.

During System Level testing, the Ω_{Wt_+}, Ω_{Wt_-} values used to compute $\underline{\omega}'$ in angular rate sensor compensation Equation (8.1.1.3-21) would be as in (8.1.1.3-22), but with the $\kappa_{Syst\,...}$ matrices set to zero. The results of the System Level testing would, of course, then be used to calculate the $\kappa_{Syst\,...}$ matrices.

To conclude the discussion on scale factor asymmetry, let us recall that the previous results for A_{Wt} and Ω_{Wt} were based on the Equation (8.1.1.2.1-15) simplified form of G_{Scal} for the accelerometers (and by extension, the Equation (8.1.1.1.1-22) simplified form of F_{Scal} for the angular rate sensors). Therefore, when applying the previous results, we should also use the corresponding simplified forms for G_{Algn}, $\underline{L}_{Bias}$, F_{Algn} and $\underline{K}_{Bias}$ that were derived from the same assumptions leading to G_{Scal}, F_{Scal} Equations (8.1.1.2.1-15) and (8.1.1.1.1-22). Thus, G_{Algn} and $\underline{L}_{Bias}$ would be calculated from Equations (8.1.1.2.1-16) - (8.1.1.2.1-17); F_{Algn} and $\underline{K}_{Bias}$ would be calculated from Equations (8.1.1.1.1-27) - (8.1.1.1.1-28). Also note, from (8.1.1.3-2) and its angular rate sensor equivalent, that the Off Diag terms in (8.1.1.2.1-16) and (8.1.1.1.1-27) are equivalently:

$$\text{Off Diag}\left(\lambda_{SystScal/Mis}\right) = \text{Off Diag}\left(\lambda_{SystScalLin/Mis}\right) \tag{8.1.1.3-23}$$

$$\text{Off Diag}\left(\kappa_{SystScal/Mis}\right) = \text{Off Diag}\left(\kappa_{SystScalLin/Mis}\right) \tag{8.1.1.3-24}$$

8.1.2 INERTIAL SENSOR INTEGRATED OUTPUT COMPENSATION ALGORITHMS

The Chapter 7 strapdown inertial navigation software algorithms are based on the use of compensated inertial sensor inputs (i.e., after application of the inertial sensor compensation equations developed in Sections 8.1.1.1 and 8.1.1.2). Specifically, compensated inertial sensor data is utilized for $\underline{\alpha}_m$ in Equations (7.1.1.1-12), for $\underline{\upsilon}_m$ in Equations (7.2.2.2-23) or (7.2.2.2-26), for $\underline{\alpha}_m$ and $\underline{\upsilon}_m$ in Equations (7.2.2.2-25) or (7.2.2.2.1-7) with (7.2.2.2.1-8), for $\underline{S}_{\upsilon_m}$ in Equations (7.3.3-9), and for $\underline{\alpha}_m$, $\underline{\upsilon}_m$, $\underline{S}_{\alpha_m}$, and $\underline{S}_{\upsilon_m}$ in Equations (7.3.3-11) or (7.3.3.1-16). Additionally, compensated sensor data based on the Section 8.1.1.1 and 8.1.1.2 equations is used in the coning, sculling and scrolling algorithms for attitude/velocity/ position updating, and in the accelerometer size-effect/anisoinertia correction algorithms. Sections 8.1.2.1 and 8.1.2.2 to follow utilize the Section 8.1.1.1 and 8.1.1.2 equations to develop compensation algorithms for the $\underline{\alpha}_m$ and $\underline{S}_{\alpha_m}$ integrated angular rate terms and for the $\underline{\upsilon}_m$ and $\underline{S}_{\upsilon_m}$ integrated specific force terms. Sensor compensation for the accelerometer size effect, accelerometer anisoinertia, coning, sculling and scrolling algorithms are developed in subsequent Sections 8.1.4.1 (and subsections), 8.1.4.2, 8.2.1.1, 8.2.2.1, 8.2.2.2 and 8.2.3.1.

It is to be noted that the Chapter 19 (Section 19.1) unified strapdown velocity/position algorithms (in Equations (19.1.5-9) with (19.1.11-1)) are also based on the use of compensated inertial sensor inputs. A compensated rotation vector $\underline{\phi}_m$ for (19.1.5-9) is derived directly from the 8.2.1.1 result by summing the compensated coning equation with the compensated $\underline{\alpha}_m$ from Section 8.1.2.1. The compensated velocity translation vector $\underline{\eta}_m$ for (19.1.5-9) is derived directly from the 8.2.2.1 result by summing the compensated sculling equation with the compensated $\underline{\upsilon}_m$ from Section 8.1.2.2 (because the sculling input $\delta\underline{\eta}_{Scul_m}$ to $\underline{\eta}_m$ in (19.1.11-1) is analytically identical to the Chapter 7 sculling term $\Delta\underline{v}_{Scul_m}$ whose compensated form is derived in 8.2.2.1). A compensated $\underline{S}_{\upsilon_m}$ for the position translation vector $\underline{\zeta}_m$ input in (19.1.11-1) is as derived in 8.1.2.2. A compensated version of the scrolling input $\delta\underline{\zeta}_{Scrol_m}$ to $\underline{\zeta}_m$ in (19.1.11-1) (for use in (19.1.5-9)) has not yet been developed as of this writing.

8.1.2.1 ANGULAR RATE SENSOR INTEGRATED OUTPUT COMPENSATION ALGORITHMS

The uncompensated form of $\underline{\alpha}_m$ and $\underline{S}_{\alpha_m}$ is obtained from the basic definition of $\underline{\alpha}_m$ and $\underline{S}_{\alpha_m}$ (e.g., Equation (7.3.3-10)), but computed from raw angular rate sensor output data:

$$\begin{aligned} &\underline{\alpha}_{Cnt}(\tau) = \int_{t_{m-1}}^{\tau} d\underline{\alpha}_{Cnt} \qquad && \underline{\alpha}_{Cnt_m} = \underline{\alpha}_{Cnt}(t_m) \\ &\underline{S}_{\alpha Cnt}(t) = \int_{t_{m-1}}^{t} \underline{\alpha}_{Cnt}(\tau)\, d\tau \qquad && \underline{S}_{\alpha Cnt_m} = \underline{S}_{\alpha Cnt}(t_m) \end{aligned} \tag{8.1.2.1-1}$$

where

$d\underline{\alpha}_{Cnt} = \underline{\omega}_{Puls}\, dt$ = Uncompensated angular rate sensor triad output differential pulse count vector.

$\underline{\omega}_{Puls}$ = Uncompensated angular rate sensor triad output pulse rate vector.

$\underline{\alpha}_{Cnt_m}$, $\underline{S}_{\alpha Cnt_m}$ = $\underline{\alpha}_m$, $\underline{S}_{\alpha_m}$ computed from uncompensated angular rate sensor triad output pulse data.

Digital integration algorithms for $\underline{\alpha}_{Cnt_m}$, $\underline{S}_{\alpha Cnt_m}$ in (8.1.2.1-1) are provided by Equations (8.2.1.1-15) and (8.2.3.1-4) in subsequent Sections 8.2.1.1 and 8.2.3.1.

The compensated form of $\underline{\alpha}_m$ and $\underline{S}_{\alpha_m}$ derives from the Equation (8.1.1.1-8) angular rate sensor triad compensation formulas repeated below:

$$\begin{aligned} \underline{\omega}' &= \Omega_{Wt}\, \underline{\omega}_{Puls} \\ \underline{\omega} &\approx \underline{\omega}' - K_{Mis}\, \underline{\omega}' - \underline{K}_{Bias} - F_{Algn}^{-1}\, \delta\underline{\omega}_{Quant} \end{aligned} \tag{8.1.2.1-2}$$

in which Ω_{Wt}, K_{Mis} and $\underline{K}_{Bias}$ are as calculated in (8.1.1.1-5) - (8.1.1.1-7).

Applying (8.1.2.1-2) to Equation (8.1.2.1-1) and approximating F_{Algn} as identity obtains the compensated $\underline{\alpha}_m$, $\underline{S}_{\alpha_m}$ algorithms:

$$\underline{\alpha}'_m = \Omega_{Wt}\, \underline{\alpha}_{Cnt_m} \tag{8.1.2.1-3}$$

$$\underline{\alpha}_m \approx \underline{\alpha}'_m - K_{Mis}\, \underline{\alpha}'_m - \underline{K}_{Bias} T_m - \delta\underline{\alpha}_{QuantC_m} \tag{8.1.2.1-4}$$

$$\underline{S}'_{\alpha_m} = \Omega_{Wt}\, \underline{S}_{\alpha Cnt_m} \tag{8.1.2.1-5}$$

$$\underline{S}_{\alpha_m} \approx \underline{S}'_{\alpha_m} - K_{Mis}\, \underline{S}'_{\alpha_m} - \frac{1}{2}\left(\underline{K}_{Bias} T_m + \delta\underline{\alpha}_{QuantC_m}\right) T_m \tag{8.1.2.1-6}$$

where

$\delta\underline{\alpha}_{QuantC_m}$ = Integrated angular rate triad quantization compensation at computer cycle m (See Section 8.1.3.3 for associated algorithms).

T_m = Computer update cycle time period.

$'$ = Reference to scaled parameters, but compensated only for scale factor error.

Equations (8.1.2.1-4) and (8.1.2.1-6) are based on the assumption that $\underline{K}_{Bias}$ can be treated as a constant over the computer update time interval m. If $\underline{K}_{Bias}$ contains dynamic time varying terms (e.g., specific force sensitive elements), a more complicated version of the $\underline{K}_{Bias}$ integral over the computer update period would be required. Also note that in Equation (8.1.2.1-6), the approximation has been made that $\delta\underline{\alpha}_{QuantC_m}$ builds linearly across the computer update interval.

Equations (8.1.2.1-3) and (8.1.2.1-5) for $\underline{\alpha}'_m$, $\underline{S}'_{\alpha_m}$ are based on the assumption that Ω_{Wt} can be approximated as constant over an m cycle. Under this assumption, if Ω_{Wt} contains non-linear terms (as addressed in Section 8.1.1.3), Ω_{Wt} would be calculated as a function of $\underline{\alpha}_{Cnt_m}$. For the more general case in which Ω_{Wt} may have rapid variations, $\underline{\alpha}'_m$, $\underline{S}'_{\alpha_m}$ can be calculated based on Equation (7.3.3-10), but computed from scale factor compensated angular rate sensor output data $\underline{\omega}'$ as in (8.1.2.1-2):

$$\underline{\alpha}'(\tau) = \int_{t_{m-1}}^{\tau} \Omega_{Wt}\, d\underline{\alpha}_{Cnt} \qquad \underline{\alpha}'_m = \underline{\alpha}'(t_m) \tag{8.1.2.1-7}$$

$$\underline{S}'_{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\alpha}'(\tau)\, d\tau \qquad \underline{S}'_{\alpha_m} = \underline{S}'_{\alpha}(t_m) \tag{8.1.2.1-8}$$

If scale factor asymmetry is to be compensated as described in Section 8.1.1.3, the $\underline{\alpha}'(\tau)$ integrand in (8.1.2.1-7) would be based on $\underline{\omega}'$ from (8.1.1.3-21) (repeated below) replacing $\underline{\omega}'$ in (8.1.2.1-2):

$$\underline{\omega}' = \Omega_{Wt_+}\, \underline{\omega}_{+Puls} + \Omega_{Wt_-}\, \underline{\omega}_{-Puls} \tag{8.1.2.1-9}$$

in which Ω_{Wt_+}, Ω_{Wt_-} are scale factor weighting matrices as defined in Equation (8.1.1.3-22) for positive and negative uncompensated angular rate sensor pulse rate output data ($\underline{\omega}_{+Puls}$ and $\underline{\omega}_{-Puls}$ - See Section 8.1.1.3 for further clarification). Then, based on (8.1.2.1-7) and (8.1.2.1-9), $\underline{\alpha}'_m$ would be calculated as:

$$\underline{\alpha}'(\tau) = \int_{t_{m-1}}^{\tau} \left(\Omega_{Wt_+}\, d\underline{\alpha}_{+Cnt} + \Omega_{Wt_-}\, d\underline{\alpha}_{-Cnt} \right) \qquad \underline{\alpha}'_m = \underline{\alpha}'(t_m) \tag{8.1.2.1-10}$$

where

$d\underline{\alpha}_{+Cnt}, d\underline{\alpha}_{-Cnt}$ = $\underline{\omega}_{+Puls}$ dt and $\underline{\omega}_{-Puls}$ dt, the uncompensated angular rate sensor triad differential positive and negative pulse rate output vectors.

Digital integration algorithms for $\underline{\alpha}'_m$ in (8.1.2.1-10) and $\underline{S}'_{\alpha_m}$ in (8.1.2.1-8) are provided by Equations (8.2.1.1-18) and (8.2.3.1-7) in subsequent Sections 8.2.1.1 and 8.2.3.1.

8.1.2.2 ACCELEROMETER INTEGRATED OUTPUT COMPENSATION ALGORITHMS

The uncompensated form of $\underline{\upsilon}_m$ and $\underline{S}_{\upsilon_m}$ is obtained from the basic definition of $\underline{\upsilon}_m$ and $\underline{S}_{\upsilon_m}$ (e.g., (7.3.3-10)), but computed from raw accelerometer output data:

$$\underline{\upsilon}_{Cnt}(\tau) = \int_{t_{m-1}}^{\tau} d\underline{\upsilon}_{Cnt} \qquad \underline{\upsilon}_{Cnt_m} = \underline{\upsilon}_{Cnt}(t_m)$$
$$\underline{S}_{\upsilon Cnt}(t) = \int_{t_{m-1}}^{t} \underline{\upsilon}_{Cnt}(\tau)\, d\tau \qquad \underline{S}_{\upsilon Cnt_m} = \underline{S}_{\upsilon Cnt}(t_m) \tag{8.1.2.2-1}$$

where

$d\underline{\upsilon}_{Cnt} = \underline{a}_{SF_{Puls}}\, dt$ = Uncompensated accelerometer triad output differential pulse count vector.

$\underline{a}_{SF_{Puls}}$ = Uncompensated accelerometer triad output pulse rate vector.

$\underline{\upsilon}_{Cnt_m}, \underline{S}_{\upsilon Cnt_m}$ = $\underline{\upsilon}_m, \underline{S}_{\upsilon_m}$ computed from uncompensated accelerometer triad output data.

Digital integration algorithms for $\underline{\upsilon}_{Cnt_m}, \underline{S}_{\upsilon Cnt_m}$ in (8.1.2.2-1) are provided by Equations (8.2.2.1-36) and (8.2.3.1-4) in subsequent Sections 8.2.2.1 and 8.2.3.1.

The compensated form of $\underline{\upsilon}_m$ and $\underline{S}_{\upsilon_m}$ derives from the Equation (8.1.1.2-8) accelerometer triad compensation formulas repeated below:

$$\underline{a}'_{SF} = A_{Wt}\, \underline{a}_{SF_{Puls}}$$
$$\underline{a}_{SF} \approx \underline{a}'_{SF} - L_{Mis}\, \underline{a}'_{SF} - \underline{L}_{Bias} - G_{Algn}^{-1}\left(\delta\underline{a}_{Size} + \delta\underline{a}_{Aniso} + \delta\underline{a}_{Quant}\right) \tag{8.1.2.2-2}$$

with A_{Wt}, L_{Mis} and $\underline{L}_{Bias}$ as calculated in (8.1.1.2-5) - (8.1.1.2-7).

Applying Equation (8.1.2.2-2) to (8.1.2.2-1) obtains the compensated $\underline{\upsilon}_m, \underline{S}_{\upsilon_m}$ algorithms:

$$\underline{\upsilon}'_m = A_{Wt}\, \underline{\upsilon}_{Cnt_m} \tag{8.1.2.2-3}$$

$$\underline{\upsilon}_m \approx \underline{\upsilon}'_m - L_{Mis}\, \underline{\upsilon}'_m - \underline{L}_{Bias}\, T_m - \delta\underline{\upsilon}_{SizeC_m} - \delta\underline{\upsilon}_{AnisoC_m} - \delta\underline{\upsilon}_{QuantC_m} \tag{8.1.2.2-4}$$

$$\underline{S}'_{\upsilon_m} = A_{Wt}\, \underline{S}_{\upsilon Cnt_m} \tag{8.1.2.2-5}$$

$$\underline{S}_{\upsilon_m} \approx \underline{S}'_{\upsilon_m} - L_{Mis}\, \underline{S}'_{\upsilon_m} - \frac{1}{2}\left(\underline{L}_{Bias} T_m + \delta\underline{\upsilon}_{SizeC_m} + \delta\underline{\upsilon}_{AnisoC_m} + \delta\underline{\upsilon}_{QuantC_m}\right) T_m \tag{8.1.2.2-6}$$

where

$\delta \underline{\upsilon}_{SizeC_m}$, $\delta \underline{\upsilon}_{AnisoC_m}$, $\delta \underline{\upsilon}_{QuantC_m}$ = Accelerometer triad size effect, anisoinertia and quantization compensations over computer cycle m (See Sections 8.1.3.3, 8.1.4.1 (and subsections) and 8.1.4.2 for associated algorithms). For the $\delta \underline{\upsilon}_{AnisoC_m}$ and $\delta \underline{\upsilon}_{QuantC_m}$ algorithms, the G_{Algn}^{-1} multiplier in (8.1.2.2-2) is approximated by identity. For the $\delta \underline{\upsilon}_{SizeC_m}$ algorithm, versions are provided that approximate G_{Algn}^{-1} by identity, and which include the G_{Algn}^{-1} multiplier.

Equations (8.1.2.2-4) and (8.1.2.2-6) are based on the assumption that $\underline{L}_{Bias}$ can be treated as a constant over the computer update time interval m. If $\underline{L}_{Bias}$ contains dynamic time varying terms (e.g., angular rate sensitive elements), a more complicated version of the $\underline{L}_{Bias}$ integral over the computer update period would be required. Also note that in Equation (8.1.2.2-6), the approximation has been made that $\delta \underline{\upsilon}_{QuantC_m}$, $\delta \underline{\upsilon}_{SizeC_m}$, $\delta \underline{\upsilon}_{AnisoC_m}$ build linearly across the computer update interval.

Equations (8.1.2.2-3) and (8.1.2.2-5) for $\underline{\upsilon}'_m$, $\underline{S}'_{\upsilon_m}$ are based on the assumption that A_{Wt} can be approximated as constant over an m cycle. Under this assumption, if A_{Wt} contains non-linear terms (as addressed in Section 8.1.1.3), A_{Wt} would be calculated as a function of $\underline{\upsilon}_{Cnt_m}$. For the more general case in which A_{Wt} may have rapid variations, $\underline{\upsilon}'_m$, $\underline{S}'_{\upsilon_m}$ can be calculated based on Equation (7.3.3-10), but computed from scale factor compensated accelerometer output data $\underline{a}'_{SF}$ as in (8.1.2.2-2):

$$\underline{\upsilon}'(\tau) = \int_{t_{m-1}}^{\tau} A_{Wt} \, d\underline{\upsilon}_{Cnt} \qquad \underline{\upsilon}'_m = \underline{\upsilon}'(t_m) \tag{8.1.2.2-7}$$

$$\underline{S}'_{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{\upsilon}'(\tau) \, d\tau \qquad \underline{S}'_{\upsilon_m} = \underline{S}'_{\upsilon}(t_m) \tag{8.1.2.2-8}$$

If scale factor asymmetry is to be compensated as described in Section 8.1.1.3, the $\underline{\upsilon}'(\tau)$ integrand in (8.1.2.2-7) would be based on $\underline{a}'_{SF}$ from (8.1.1.3-19) (repeated below) replacing $\underline{a}'_{SF}$ in (8.1.2.2-2):

$$\underline{a}'_{SF} = A_{Wt_+} \underline{a}_{SF_{+Puls}} + A_{Wt_-} \underline{a}_{SF_{-Puls}} \tag{8.1.2.2-9}$$

in which A_{Wt_+}, A_{Wt_-} are scale factor weighting matrices as defined in Equation (8.1.1.3-20) for positive and negative uncompensated accelerometer pulse rate output data ($\underline{a}_{SF_{+Puls}}$ and

$\underline{a}_{SF\text{-}Puls}$ - See Section 8.1.1.3 for further clarification). Then, based on (8.1.2.2-7) and (8.1.2.2-9), $\underline{\upsilon}'_m$ would be calculated as:

$$\underline{\upsilon}'(\tau) = \int_{t_{m-1}}^{\tau} \left(A_{Wt_+}\, d\underline{\upsilon}_{+Cnt} + A_{Wt_-}\, d\underline{\upsilon}_{-Cnt} \right) \qquad \underline{\upsilon}'_m = \underline{\upsilon}'(t_m) \qquad (8.1.2.2\text{-}10)$$

where

$d\underline{\upsilon}_{+Cnt}, d\underline{\upsilon}_{-Cnt}$ = $\underline{a}_{SF+Puls}\,dt$ and $\underline{a}_{SF\text{-}Puls}\,dt$, the uncompensated accelerometer triad differential positive and negative pulse rate output vectors.

Digital integration algorithms for $\underline{\upsilon}'_m$ in (8.1.2.2-10) and $\underline{S}'_{\upsilon_m}$ in (8.1.2.2-8) are provided by Equations (8.2.2.1-39) and (8.2.3.1-7) in subsequent Sections 8.2.2.1 and 8.2.3.1.

8.1.3 INERTIAL SENSOR QUANTIZATION COMPENSATION ALGORITHMS

The design of a strapdown inertial sensor (angular rate sensor or accelerometer) and its computer interface electronics typically includes a pulse output function and an integration function. The pulse output function generates an output logic pulse each time the integrated input to the sensor reaches a specific value known as the pulse size. A “positive” logic pulse is output when the integrated input (since the last output pulse) is positive and equal to the pulse size; a “negative” pulse is output when the integrated input (since the last pulse output) is negative and equal to the pulse size. The integration function is implemented by the pulse generation operation and by counting the plus and minus output pulses provided to the strapdown navigation computer, thereby accurately reconstructing the integral of the sensor input (within a pulse). Without compensation, the accuracy of the reconstructed integral in the strapdown computer is limited to the pulse size. The associated error is denoted as “Pulse Quantization Error” and the associated compensation will be defined as “Quantization Compensation”.

One of two types of quantization compensation can be utilized to reduce pulse quantization error, depending on the particular sensor utilized and its output availability; Pulse Count Residual compensation or Turn-Around Dead-Band compensation.

8.1.3.1 PULSE COUNT RESIDUAL COMPENSATION

For some inertial sensors, the integrated input minus the sum of the pulses emitted (“pulse count residual”) is a measurable analog signal and available as an additional output (sometimes including a bias). For other inertial sensors, the pulse count residual is estimated as the ratio of the measured time interval from the previous pulse to the sensor data sample time, divided by the current measured time between pulses (Note - To assure a reasonably short time interval between pulses, time base measurement of the pulse count residual is usually restricted to sensors having a biased input). Pulse Count Residual compensation utilizes the pulse count residual output to refine the pulse count sample in the strapdown computer. The pulse count residual is created by the pulse generation logic within the inertial sensor electronics and can be defined analytically as:

$$\mathrm{CntRes} = \mathrm{CntRes}_0 + \frac{1}{\mathrm{Wt}}\,\mathrm{Integ} - \mathrm{Cnt} \qquad (8.1.3.1\text{-}1)$$

where

CntRes = Pulse count residual output since the end of the last computer cycle l (i.e., since the end of cycle l-1).

CntRes_0 = Value of CntRes at the start of the first l cycle.

Integ = Integrated sensor input from the start of the first l cycle to the current time within the current l cycle (measured in radians for an integrating angular rate sensor and ft per sec for an accelerometer).

Cnt = Sensor output pulse count from the start of the first l cycle to the current time within the current l cycle.

Wt = Pulse weighting coefficient for the particular sensor (measured in rad per pulse for an angular rate sensor and ft per sec per pulse for an accelerometer).

At the end of cycle l immediately following the emission of the last pulse associated with cycle l, Equation (8.1.3.1-1) is:

$$\mathrm{CntRes}_l = \mathrm{CntRes}_0 + \frac{1}{\mathrm{Wt}}\,\mathrm{Integ}_l - \mathrm{Cnt}_l \qquad (8.1.3.1\text{-}2)$$

where

CntRes_l = Pulse count residual output from the inertial sensor at the end of computer cycle l.

Integ_l = Integrated sensor input from the start of the first l cycle to the end of the current l cycle.

Cnt_l = Sensor output pulse count from the start of the first l cycle to the end of the current l cycle.

The difference between pulse count residual measurements over two successive l cycles is the difference between (8.1.3.1-2) at l and l-1:

$$\mathrm{CntRes}_l - \mathrm{CntRes}_{l-1} = \frac{1}{\mathrm{Wt}}\left(\mathrm{Integ}_l - \mathrm{Integ}_{l-1}\right) - \left(\mathrm{Cnt}_l - \mathrm{Cnt}_{l-1}\right) \qquad (8.1.3.1\text{-}3)$$

With rearrangement and redefinition of terms, (8.1.3.1-3) is equivalently:

$$\Delta \mathrm{Integ}_l \;=\; \mathrm{Wt}\left(\Delta \mathrm{Cnt}_l + \mathrm{CntRes}_l - \mathrm{CntRes}_{l-1}\right) \tag{8.1.3.1-4}$$

where

$\Delta \mathrm{Integ}_l$ = Integrated sensor input over computer cycle l.

$\Delta \mathrm{Cnt}_l$ = Sensor output pulse count over computer cycle l.

The quantization error is defined as:

$$\delta \mathrm{Quant}_l \;\equiv\; \mathrm{Wt}\,\Delta \mathrm{Cnt}_l - \Delta \mathrm{Integ}_l \tag{8.1.3.1-5}$$

or, with (8.1.3.1-4):

$$\delta \mathrm{Quant}_l \;=\; -\,\mathrm{Wt}\left(\mathrm{CntRes}_l - \mathrm{CntRes}_{l-1}\right) \tag{8.1.3.1-6}$$

where

$\delta \mathrm{Quant}_l$ = Quantization error over cycle l.

Equation (8.1.3.1-6) is the quantization compensation correction for application each computer l cycle. It is compatible with the definitions of the quantization compensation terms in Equations (8.1.2.1-2) - (8.1.2.1-4) ($\delta \underline{\alpha}_{\mathrm{QuantC_m}}$ for angular rate sensors) and Equations (8.1.2.2-2) - (8.1.2.2-4) ($\delta \underline{\upsilon}_{\mathrm{QuantC_m}}$ for accelerometers), but for an l cycle within an m cycle. Over an m cycle, the equivalent to (8.1.3.1-6) would be:

$$\delta \mathrm{Quant}_m \;=\; -\,\mathrm{Wt} \sum_{l\,=\,(m-1)\,k}^{m\,k} \left(\mathrm{CntRes}_l - \mathrm{CntRes}_{l-1}\right) \;=\; -\,\mathrm{Wt}\left(\mathrm{CntRes}_m - \mathrm{CntRes}_{m-1}\right) \tag{8.1.3.1-7}$$

where

CntRes_m = Pulse count residual output from the inertial sensor sampled at the end of each computer cycle m, representing the integrated input since the last pulse output (measured in fractions of a pulse).

$\delta \mathrm{Quant}_m$ = Quantization error over cycle m.

k = Number of l computer cycles in one m computer cycle.

Equation (8.1.3.1-7) is the quantization compensation correction form for application once each computer m cycle.

8.1.3.2 TURN-AROUND DEAD-BAND COMPENSATION

For an idealized inertial sensor (an angular rate sensor for example), output pulses are emitted consecutively on a plus or minus output line (depending on the sign of the input rate), with each pulse representing a fixed angular movement through an integrated angular rate increment equal to the sensor pulse size. A helpful analogy for understanding the pulse output logic is to imagine traveling past a picket fence. The picket fence represents inertial space, and the distance between the right edge of two successive pickets corresponds to the sensor pulse size. Imagine observing the picket fence while looking through a small viewing window (representing the inertial sensor) that restricts your field of view to no more than one picket at a time. As the window passes each picket, a pulse is issued on a plus or minus output line, depending on whether the window is moving from left-to-right or right-to-left past the picket in view. By summing the pulses, the net distance traveled past the picket fence (the integrated output from the inertial sensor) can be determined.

Let us assume that the sensor pulse output logic is designed to emit an output pulse at the instant the right edge of the viewing window passes the right edge of the picket in view (the "trigger point"). Consider a condition for which the sensor is rotating in the positive direction, and the last sensor output pulse was emitted at a trigger point corresponding to the sensor being at integrated rate angle θ (around its input axis). Now assume that at angle $\theta + \Delta\theta$ the sensor reverses direction. Let us further assume that $\Delta\theta$ (a positive value) is smaller than the sensor pulse size so that no additional positive output pulses are triggered. Following the change in direction, the ideal sensor would rotate negatively through minus $\Delta\theta$ and emit a negative pulse at the instant its pulse trigger edge passed the trigger point, again at the angle equal to θ condition. The negative pulse would thereby exactly cancel the positive pulse emitted previously when passing by θ under positive rate. Now consider that the previous rotation sequence occurred for a very slow positive rate through θ to $\theta + \Delta\theta$, and a very slow turn around back through θ negatively. In the presence of electrical noise in the pulse output circuitry or for high frequency low amplitude input rate, flurries of plus and minus outputs would be emitted as the sensor crossed angle θ in the positive direction, and again as it re-crossed negatively. To avoid this condition, a dead-band is typically built into the sensor pulse output logic that will only issue the first negative output pulse after the sensor has rotated negatively through a specified fixed angle increment (dead-band) past the θ point when the last positive pulse was emitted. Reciprocal logic is also included for rate reversals from minus to plus. The above discussion applies equally to accelerometers for which the dead-band in the pulse logic represents a small integrated specific force acceleration increment on the order of the accelerometer pulse size in magnitude.

An analytical representation of the above pulse generation process can be developed as follows. A positive pulse is emitted when CntRes as defined in Equation (8.1.3.1-1) exceeds a specified positive trigger threshold. Simultaneously, CntRes is reduced by one (in accordance with Equation (8.1.3.1-1)). For continuing positive motion, the Integ term in (8.1.3.1-1) builds up positively until CntRes again reaches the positive trigger threshold, another positive pulse is emitted, and CntRes is reduced again by one. The net effect under continuing positive motion is to control CntRes so that its average value equals the positive trigger threshold value minus one half. The reciprocal effect applies for continuous negative motion; a negative trigger threshold is used to trigger negative pulses, and the CntRes average value is controlled to equal the negative trigger threshold value plus one half. As described above, when a positive pulse is emitted, CntRes is reduced by one. The added dead-band logic described in the previous paragraph then requires the Integ term in (8.1.3.1-1) to be further reduced by the selected dead-band value before a negative pulse is issued. The result equates the difference between the positive and negative trigger thresholds to one plus the dead-band. Stated differently, the negative trigger threshold equals the positive trigger threshold minus the quantity, one plus the dead-band. The previous description can be described analytically as:

$$\begin{aligned}
\mathrm{CntRes}_{Avg} &= \left(\mathrm{Thrsh}_{+} - \frac{1}{2}\right)\mathrm{Step}\left(\mathrm{Puls}_{last}\right) + \left(\mathrm{Thrsh}_{-} + \frac{1}{2}\right)\left[1 - \mathrm{Step}\left(\mathrm{Puls}_{last}\right)\right] \\
&= \left(\mathrm{Thrsh}_{+} - \frac{1}{2}\right)\mathrm{Step}\left(\mathrm{Puls}_{last}\right) \\
&\qquad + \left(\mathrm{Thrsh}_{+} - (1 + \mathrm{db}) + \frac{1}{2}\right)\left[1 - \mathrm{Step}\left(\mathrm{Puls}_{last}\right)\right] \\
&= \left[\mathrm{Thrsh}_{+} - \frac{1}{2} - \left(\mathrm{Thrsh}_{+} - (1 + \mathrm{db}) + \frac{1}{2}\right)\right]\mathrm{Step}\left(\mathrm{Puls}_{last}\right) \\
&\qquad + \mathrm{Thrsh}_{+} - (1 + \mathrm{db}) + \frac{1}{2} \\
&= \mathrm{Thrsh}_{+} - \frac{1}{2} - \mathrm{db} + \mathrm{db}\ \mathrm{Step}\left(\mathrm{Puls}_{last}\right)
\end{aligned} \tag{8.1.3.2-1}$$

where

CntRes_{Avg} = Average value of CntRes.

Thrsh_{+}, Thrsh_{-} = Positive and negative trigger threshold value.

Puls_{last} = Value of the last pulse emitted (+1 or -1).

Step ( ) = Unit step function having a value of zero for ( ) < 0 and unity for ( ) ≥ 0.

We also note that:

$$\mathrm{Step}\,(\,) = \frac{1}{2}\left[1 + \mathrm{Sign}\,(\,)\right] \tag{8.1.3.2-2}$$

where

Sign ( ) = +1 for ( ) ≥ 0 and -1 for ( ) < 0.

With (8.1.3.2-2), Equation (8.1.3.2-1) becomes:

$$\mathrm{CntRes}_{Avg} = \mathrm{Thrsh}_{+} - \frac{1}{2} - \frac{1}{2}\,\mathrm{db} + \frac{1}{2}\,\mathrm{db}\ \mathrm{Sign}\left(\mathrm{Puls}_{last}\right) \tag{8.1.3.2-3}$$

Equation (8.1.3.2-3) accurately represents the average value of CntRes during continuous positive or negative motion (i.e., when the current and last pulse output are of the same sign). During the turn-around transition from positive to negative or negative to positive motion (8.1.3.2-1) is in error. However, for a sensor with a small enough pulse size and fast sampling rate (compared to expected high frequency dynamic motion inputs), the fraction of the time in turn-arounds will be small compared to the time for motion in one direction (i.e., motion over two successive pulses). On this basis, the error in (8.1.3.2-1) during turn-arounds can be considered small, hence, negligible. Turn-Around Dead-Band compensation is based on using $\mathrm{CntRes}_{\mathrm{Avg}}$ as defined by (8.1.3.2-3) as an approximation to the actual CntRes signal. Thus, at the end of an l cycle we approximate CntRes from (8.1.3.2-3) as:

$$\mathrm{CntRes}_l \approx \mathrm{Thrsh}_+ - \frac{1}{2} - \frac{1}{2}\,\mathrm{db} + \frac{1}{2}\,\mathrm{db}\ \mathrm{Sign}\left(\mathrm{Puls}_l\right) \tag{8.1.3.2-4}$$

where

Puls_l = Value for the pulse emitted immediately preceding the end of cycle l.

The difference between (8.1.3.2-4) over two successive l cycles is:

$$\mathrm{CntRes}_l - \mathrm{CntRes}_{l-1} = \frac{\mathrm{db}}{2}\left[\mathrm{Sign}\left(\mathrm{Puls}_l\right) - \mathrm{Sign}\left(\mathrm{Puls}_{l-1}\right)\right] \tag{8.1.3.2-5}$$

Turn-Around Dead-Band compensation applies Equation (8.1.3.2-5) in (8.1.3.1-6). Thus:

$$\delta\mathrm{Quant}_l = -\,\mathrm{Wt}\,\frac{\mathrm{db}}{2}\left[\mathrm{Sign}\left(\mathrm{Puls}_l\right) - \mathrm{Sign}\left(\mathrm{Puls}_{l-1}\right)\right] \tag{8.1.3.2-6}$$

or if we further approximate the Puls terms by the equivalent $\Delta\mathrm{Cnt}$ values:

$$\delta\mathrm{Quant}_l \approx -\,\mathrm{Wt}\,\frac{\mathrm{db}}{2}\left[\mathrm{Sign}\left(\Delta\mathrm{Cnt}_l\right) - \mathrm{Sign}\left(\Delta\mathrm{Cnt}_{l-1}\right)\right] \tag{8.1.3.2-7}$$

Equation (8.1.3.2-7) is compatible with the definitions of the quantization compensation terms in Equations (8.1.2.1-2) - (8.1.2.1-4) ($\delta\underline{\alpha}_{\mathrm{QuantC_m}}$ for angular rate sensors) and Equations (8.1.2.2-2) - (8.1.2.2-4) ($\delta\underline{\upsilon}_{\mathrm{QuantC_m}}$ for accelerometers), but for an l cycle within an m cycle. Over an m cycle, the equivalent to (8.1.3.2-7) would be:

$$\begin{aligned}\delta\mathrm{Quant}_m &= -\,\mathrm{Wt}\,\frac{\mathrm{db}}{2}\sum_{l=(m-1)\,k}^{m\,k}\left[\mathrm{Sign}\left(\Delta\mathrm{Cnt}_l\right) - \mathrm{Sign}\left(\Delta\mathrm{Cnt}_{l-1}\right)\right] \\ &= -\,\mathrm{Wt}\,\frac{\mathrm{db}}{2}\left[\mathrm{Sign}\left(\Delta\mathrm{Cnt}_{l=m\,k}\right) - \mathrm{Sign}\left(\Delta\mathrm{Cnt}_{l=(m-1)\,k-1}\right)\right] \\ &\approx -\,\mathrm{Wt}\,\frac{\mathrm{db}}{2}\left[\mathrm{Sign}\left(\Delta\mathrm{Cnt}_m\right) - \mathrm{Sign}\left(\Delta\mathrm{Cnt}_{m-1}\right)\right]\end{aligned} \tag{8.1.3.2-8}$$

where

$\Delta Cnt_{l = m\,k}$, $\Delta Cnt_{l = (m-1)\,k\,-\,1}$ = ΔCnt_l over computer l cycles mk and (m-1)k-1.

ΔCnt_m, ΔCnt_{m-1} = Sensor output pulse counts over computer cycles m and m-1.

with $\delta Quant_m$ and k as defined in Section 8.1.3.1.

Equations (8.1.3.2-7) and (8.1.3.2-8) account for the average error introduced at turn-around by the pulse dead-band logic.

8.1.3.3 PULSE QUANTIZATION COMPENSATION ALGORITHM FORMS

The $\delta\underline{\alpha}_{QuantC_m}$, $\delta\underline{\upsilon}_{QuantC_m}$, $\delta\underline{\alpha}_{QuantC_{l\,:\,m}}$ pulse quantization terms included in Equations (8.1.2.1-4), (8.1.2.1-6), (8.1.2.2-4), (8.1.2.2-6), and subsequently (8.1.4.1.4-10), represent implementations of Equations (8.1.3.1-6) and (8.1.3.1-7), or (8.1.3.2-7) and (8.1.3.2-8), into the angular rate sensor and accelerometer compensation formulas. Values for the quantization compensation terms derived from Equations (8.1.3.1-6) and (8.1.3.1-7) are as follows:

$$\begin{aligned} \delta\underline{\alpha}_{QuantC_m} &= - \Omega_{Wt} \sum_{l = (m-1)\,k}^{m\,k} \left(\underline{\alpha}_{CntRes_l} - \underline{\alpha}_{CntRes_{l-1}}\right) \\ &= - \Omega_{Wt} \left(\underline{\alpha}_{CntRes_m} - \underline{\alpha}_{CntRes_{m-1}}\right) \end{aligned} \tag{8.1.3.3-1}$$

$$\delta\underline{\alpha}_{QuantC_{l\,:\,m}} = - \Omega_{Wt} \left(\underline{\alpha}_{CntRes_m} - \underline{\alpha}_{CntRes_{(l\,:\,m)\,-1}}\right) \tag{8.1.3.3-2}$$

$$\begin{aligned} \delta\underline{\upsilon}_{QuantC_m} &= - A_{Wt} \sum_{l = (m-1)\,k}^{m\,k} \left(\underline{\upsilon}_{CntRes_l} - \underline{\upsilon}_{CntRes_{l-1}}\right) \\ &= - A_{Wt} \left(\underline{\upsilon}_{CntRes_m} - \underline{\upsilon}_{CntRes_{m-1}}\right) \end{aligned} \tag{8.1.3.3-3}$$

where

$\underline{\alpha}_{CntRes_l}$, $\underline{\upsilon}_{CntRes_l}$ = Vectors with elements equal to the X, Y and Z angular rate sensor and accelerometer analog pulse count residual outputs sampled at the end of computer cycle *l*.

$\underline{\alpha}_{CntRes_m}$, $\underline{\upsilon}_{CntRes_m}$ = Vectors with elements equal to the X, Y and Z angular rate sensor and accelerometer analog pulse count residual outputs sampled at the end of computer cycle m.

$\underline{\alpha}_{CntRes_{(l\,:\,m)\,-1}}$ = $\underline{\alpha}_{CntRes_l}$ for computer l cycle immediately preceding $\underline{\alpha}_{CntRes_m}$.

Values for the quantization compensation terms derived from Equations (8.1.3.2-7) and (8.1.3.2-8) are given by:

$$
\begin{aligned}
\delta\underline{\alpha}_{QuantC_m} &= - \Omega_{Wt} \frac{db_\omega}{2} \sum_{l = (m-1)\,k}^{m\,k} \left[\mathrm{Sign}\left(\Delta\underline{\alpha}_{Cnt_l}\right) - \mathrm{Sign}\left(\Delta\underline{\alpha}_{Cnt_{l-1}}\right) \right] \\
&\approx - \Omega_{Wt} \frac{db_\omega}{2} \left[\mathrm{Sign}\left(\underline{\alpha}_{Cnt_m}\right) - \mathrm{Sign}\left(\underline{\alpha}_{Cnt_{m-1}}\right) \right]
\end{aligned} \tag{8.1.3.3-4}
$$

$$
\delta\underline{\alpha}_{QuantC_{l\,:\,m}} = - \Omega_{Wt} \frac{db_\omega}{2} \left[\mathrm{Sign}\left(\Delta\underline{\alpha}_{Cnt_m}\right) - \mathrm{Sign}\left(\Delta\underline{\alpha}_{Cnt_{(l\,:\,m)\,-1}}\right) \right] \tag{8.1.3.3-5}
$$

$$
\begin{aligned}
\delta\underline{\upsilon}_{QuantC_m} &= - A_{Wt} \frac{db_a}{2} \sum_{l = (m-1)\,k}^{m\,k} \left[\mathrm{Sign}\left(\Delta\underline{\upsilon}_{Cnt_l}\right) - \mathrm{Sign}\left(\Delta\underline{\upsilon}_{Cnt_{l-1}}\right) \right] \\
&\approx - A_{Wt} \frac{db_a}{2} \left[\mathrm{Sign}\left(\underline{\upsilon}_{Cnt_m}\right) - \mathrm{Sign}\left(\underline{\upsilon}_{Cnt_{m-1}}\right) \right]
\end{aligned} \tag{8.1.3.3-6}
$$

where

db_ω, db_a = Angular rate sensor and accelerometer pulse output turn-around logic dead-bands.

$\Delta\underline{\alpha}_{Cnt_l}$, $\Delta\underline{\upsilon}_{Cnt_l}$ = Pulse output summation vectors with elements equal for the X, Y, Z angular rate sensor and accelerometer pulse counts over computer cycle *l*.

$\Delta\underline{\alpha}_{Cnt_m}$, $\Delta\underline{\upsilon}_{Cnt_m}$ = $\Delta\underline{\alpha}_{Cnt_l}$, $\Delta\underline{\upsilon}_{Cnt_l}$ at the end of computer cycle m.

$\Delta\underline{\alpha}_{Cnt_{(l\,:\,m)\,-1}}$ = $\Delta\underline{\alpha}_{Cnt_l}$ for computer *l* cycle immediately preceding $\Delta\underline{\alpha}_{Cnt_m}$.

$\underline{\alpha}_{Cnt_m}$, $\underline{\upsilon}_{Cnt_m}$ = Pulse output summation vectors with elements equal to the X, Y and Z angular rate sensor and accelerometer pulse counts over computer cycle m.

Sign ($\underline{V}$) = Vector with components Sign (V_X), Sign (V_Y), Sign (V_Z) in which V_X, V_Y, V_Z equal the X, Y, Z components of $\underline{V}$.

An alternate version of (8.1.3.3-4) - (8.1.3.3-6) can also be written based on scaled sensor data in which the bracketed terms in the Sign ( ) functions would be replaced, based on Equations (8.1.2.1-10) and (8.1.2.2-10), by the equivalent ′ scaled sensor data parameters:

$$
\delta\underline{\alpha}_{QuantC_m} \approx - \Omega_{Wt} \frac{db_\omega}{2} \left[\mathrm{Sign}\left(\underline{\alpha}'_m\right) - \mathrm{Sign}\left(\underline{\alpha}'_{m-1}\right) \right] \tag{8.1.3.3-7}
$$

$$
\delta\underline{\alpha}_{QuantC_{l\,:\,m}} = - \Omega_{Wt} \frac{db_\omega}{2} \left[\mathrm{Sign}\left(\Delta\underline{\alpha}'_m\right) - \mathrm{Sign}\left(\Delta\underline{\alpha}'_{(l\,:\,m)\,-1}\right) \right] \tag{8.1.3.3-8}
$$

$$\delta\underline{\upsilon}_{QuantC_m} \approx - A_{Wt} \frac{db_a}{2} \left[\mathrm{Sign}\left(\underline{\upsilon}'_m\right) - \mathrm{Sign}\left(\underline{\upsilon}'_{m-1}\right) \right] \tag{8.1.3.3-9}$$

where

$\Delta\underline{\alpha}'_m = \Delta\underline{\alpha}'_l$ at the end of computer cycle m.

$\Delta\underline{\alpha}'_{(l:\,m)\,-1} = \Delta\underline{\alpha}'_l$ for computer l cycle immediately preceding $\Delta\underline{\alpha}'_m$.

The $'$ scaled sensor data parameters in (8.1.3.3-7) - (8.1.3.3-9) are as calculated in subsequent Equations (8.1.4.1.4-5), (8.2.1.1-18) and (8.2.2.1-39).

If scale factor asymmetry is being compensated explicitly as in Section 8.1.1.3, the A_{Wt} and Ω_{Wt} matrices in the previous expressions can be approximated as the average of the Section 8.1.1.3 positive and negative weighting matrices, i.e.:

$$A_{Wt} \approx \frac{1}{2}\left(A_{Wt_+} + A_{Wt_-}\right) \tag{8.1.3.3-10}$$

$$\Omega_{Wt} \approx \frac{1}{2}\left(\Omega_{Wt_+} + \Omega_{Wt_-}\right) \tag{8.1.3.3-11}$$

8.1.4 ACCELEROMETER SIZE EFFECT AND ANISOINERTIA COMPENSATION ALGORITHMS

In addition to the inertial sensor corrections discussed in Sections 8.1.1 - 8.1.3, compensation is frequently employed in strapdown inertial navigation systems for accelerometer size effect and, for pendulous accelerometers, anisoinertia error. Due to the physical size of accelerometers, each accelerometer in the accelerometer triad of a strapdown inertial navigation sensor assembly cannot be collocated. As a result, each accelerometer measures the acceleration of a point at its center of seismic mass (the effective acceleration measurement point within the accelerometer), which differs from the other accelerometer measurement points. This net acceleration vector measurement error due to non-collocated accelerometers is known as accelerometer size effect error. Anisoinertia error in pendulous accelerometers is generated by reaction torque about the hinge axis proportional to the product of angular rates about the pendulum and input axis, multiplied by the difference in moments of inertia about the pendulum and input axes (Reference 31). Unlike the Sensor and System Level corrections discussed in Sections 8.1.1 - 8.1.2 whose compensation coefficients depend on individual sensor measurements, once the size effect and anisoinertia error coefficients are determined for a particular sensor model and inertial sensor assembly design, they are the same for all accelerometers and sensor assemblies of the same design.

8.1.4.1 ACCELEROMETER SIZE EFFECT COMPENSATION ALGORITHM

An analytical description of accelerometer size effect error can be derived from the relationship between the position location of an accelerometer in the sensor assembly and the position of a defined navigation reference point within the navigation system:

$$\underline{R}_k^I = \underline{R}_{Ref}^I + \underline{l}_k^I \tag{8.1.4.1-1}$$

where

I = Non-rotating inertial coordinate frame as defined in Section 2.2.

$\underline{R}_k^I$ = Position vector from earth's center to accelerometer k, projected on I Frame axes.

$\underline{R}_{Ref}^I$ = Position vector from earth's center to the INS navigation reference point, projected on I Frame axes. The navigation reference point is defined as a selected fixed point within the INS whose position and velocity is to be calculated by the inertial navigation integration algorithms.

$\underline{l}_k^I$ = Position vector from the INS navigation reference point to the accelerometer k center of seismic mass, as projected on I Frame axes.

Taking the second derivative of (8.1.4.1-1) gives:

$$\ddot{\underline{R}}_k^I = \ddot{\underline{R}}_{Ref}^I + \ddot{\underline{l}}_k^I \tag{8.1.4.1-2}$$

Equating the $\underline{R}$ derivative terms in (8.1.4.1-2) to the sum of the specific force and gravitational accelerations obtains:

$$\ddot{\underline{R}}_k^I = \underline{a}_{SF_k}^I + \underline{g}_k^I \qquad \ddot{\underline{R}}_{Ref}^I = \underline{a}_{SF_{Ref}}^I + \underline{g}_{Ref}^I \tag{8.1.4.1-3}$$

where

$\underline{a}_{SF_k}^I, \underline{a}_{SF_{Ref}}^I, \underline{g}_k^I, \underline{g}_{Ref}^I$ = Specific force and gravitational accelerations at the $\underline{R}_k^I$ and $\underline{R}_{Ref}^I$ locations. Accelerometer k measures the component of $\underline{a}_{SF_k}$ along its input axis.

The $\ddot{\underline{l}}_k^I$ term in (8.1.4.1-2) can be expressed as a function of the INS angular rate by first applying generalized Equation (3.4-6) to the $\underline{l}_k^I$ first derivative while recognizing that $\underline{l}_k$ is fixed in sensor assembly axes:

$$\dot{\underline{l}}_k^I = C_B^I \, \dot{\underline{l}}_k^B + \underline{\omega}_{IB}^I \times \underline{l}_k^I = \underline{\omega}_{IB}^I \times \underline{l}_k^I \tag{8.1.4.1-4}$$

where

$\underline{\omega}_{IB}^{I}$ = Angular rate of the sensor assembly B Frame (defined in Section 2.2) relative to inertial space, expressed in I Frame coordinates.

C_B^I = Direction cosine matrix that transforms vectors from the B Frame to the I Frame.

Taking the derivative of (8.1.4.1-4) and substituting (8.1.4.1-4) for $\dot{\underline{l}}_k^I$ yields the desired expression for the (8.1.4.1-2) $\ddot{\underline{l}}_k^I$ term:

$$\ddot{\underline{l}}_k^I = \dot{\underline{\omega}}_{IB}^I \times \underline{l}_k^I + \underline{\omega}_{IB}^I \times \dot{\underline{l}}_k^I = \dot{\underline{\omega}}_{IB}^I \times \underline{l}_k^I + \underline{\omega}_{IB}^I \times \left(\underline{\omega}_{IB}^I \times \underline{l}_k^I\right) \tag{8.1.4.1-5}$$

Substituting (8.1.4.1-5) and (8.1.4.1-3) into (8.1.4.1-2) with the approximation that the gravitational acceleration at the accelerometer and INS reference points are equal, finds after rearrangement:

$$\begin{aligned} \underline{a}_{SF_k}^I &= \underline{a}_{SF_{Ref}}^I + \underline{g}_{Ref}^I - \underline{g}_k^I + \dot{\underline{\omega}}_{IB}^I \times \underline{l}_k^I + \underline{\omega}_{IB}^I \times \left(\underline{\omega}_{IB}^I \times \underline{l}_k^I\right) \\ &\approx \underline{a}_{SF_{Ref}}^I + \dot{\underline{\omega}}_{IB}^I \times \underline{l}_k^I + \underline{\omega}_{IB}^I \times \left(\underline{\omega}_{IB}^I \times \underline{l}_k^I\right) \end{aligned} \tag{8.1.4.1-6}$$

Finally, we transform Equation (8.1.4.1-6) to the B Frame, find its component along the accelerometer k input axis, and drop the superscript and IB subscript notation for simplicity. The result is an equation for the accelerometer output as a function of the reference point specific force:

$$a_{SF_k} = \underline{G}_{Algn_k}^T \cdot \left[\underline{a}_{SF_{Ref}} + \dot{\underline{\omega}} \times \underline{l}_k + \underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k\right)\right] \tag{8.1.4.1-7}$$

where

$\underline{\omega}$ = B Frame components of the accelerometer triad inertial angular rotation rate $\underline{\omega}_{IB}$ measured by the INS strapdown angular rate sensors.

a_{SF_k} = k axis accelerometer sensed acceleration.

$\underline{a}_{SF_{Ref}}$ = B Frame components of the INS navigation reference point specific force acceleration, which is the $\underline{a}_{SF}$ vector used for INS velocity determination (as in continuous form Equation (4.3-18) with (4.2-1) and (4.2-3)).

$\underline{G}_{Algn\,k}^T$ = Vector formed from the k[th] column of G_{Algn}^T, the transpose of the G_{Algn} accelerometer triad alignment matrix. From the definition of G_{Algn} following Equation (8.1.1.2-1), the $\underline{G}_{Algn\,k}^T$ vector represents a unit vector in B Frame coordinates along the accelerometer k input axis.

The acceleration vector sensed by an orthogonal accelerometer triad, then, is from (8.1.4.1-7):

$$\begin{aligned}
\underline{a}_{Accl} &= \sum_{k=1,3} a_{SF_k}\,\underline{u}_k = \sum_{k=1,3} \left\{ \underline{G}^T_{Algn_k} \cdot \left[\underline{a}_{SF_{Ref}} + \dot{\underline{\omega}} \times \underline{l}_k + \underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k \right) \right] \right\} \underline{u}_k \\
&= \sum_{k=1,3} \left(\underline{G}^T_{Algn_k} \cdot \underline{a}_{SF_{Ref}} \right) \underline{u}_k + \sum_{k=1,3} \left\{ \underline{G}^T_{Algn_k} \cdot \left[\dot{\underline{\omega}} \times \underline{l}_k + \underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k \right) \right] \right\} \underline{u}_k \qquad (8.1.4.1\text{-}8) \\
&= G_{Algn}\,\underline{a}_{SF_{Ref}} + \sum_{k=1,3} \left\{ \underline{G}^T_{Algn_k} \cdot \left[\dot{\underline{\omega}} \times \underline{l}_k + \underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k \right) \right] \right\} \underline{u}_k
\end{aligned}$$

where

$\underline{a}_{Accl}$ = Vector formed from the three a_{SF_k} accelerometer inputs.

$\underline{u}_k$ = B Frame components of a unit vector along the nominal accelerometer k input axis. For this development we will assume that the accelerometers are nominally aligned with B Frame axes so that $\underline{u}_k$ is a unit vector along B Frame axis X, Y or Z for k = 1, 2 or 3.

For an idealized accelerometer of infinitesimal physical size, the three accelerometers in the sensor assembly can be collocated at the same physical point in the sensor assembly which would then be identified as the INS reference point. Then the $\underline{l}_k$'s would be zero and Equation (8.1.4.1-8) would reduce to $\underline{a}_{Accl} = G_{Algn}\,\underline{a}_{SF_{Ref}}$ (Note that this is of the same form as the $G_{Algn}\,\underline{a}_{SF}$ term in (8.1.1.2-1)). Because of the finite size of the actual accelerometers, the $\underline{l}_k$'s are non-zero, and $\underline{a}_{Accl} \neq G_{Algn}\,\underline{a}_{SF_{Ref}}$. The $\underline{l}_k$ terms in (8.1.4.1-8) are identified as the "size effect" error $\delta\underline{a}_{Size}$ in Equation (8.1.1.2-1).

Based on (8.1.4.1-8), the $\underline{a}_{SF_{Ref}}$ specific force acceleration of the sensor assembly reference point can be accurately computed if the $\underline{a}_{Accl}$ outputs are corrected for size effect as in (8.1.1.2-4):

$$\begin{aligned}
\underline{a}_{SF_{Ref}} &= G^{-1}_{Algn} \left(\underline{a}_{Accl} - \delta\underline{a}_{Size} \right) \\
\delta\underline{a}_{Size} &\equiv \sum_{k=1,3} \left\{ \underline{G}^T_{Algn_k} \cdot \left[\dot{\underline{\omega}} \times \underline{l}_k + \underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k \right) \right] \right\} \underline{u}_k
\end{aligned} \qquad (8.1.4.1\text{-}9)$$

where

$\delta\underline{a}_{Size}$ = Size effect correction.

With rotation compensation and sculling terms applied (e.g., as in Equations (7.2.2.2-22) - (7.2.2.2-25)) for transformation algorithm computation rate correction, the composite specific force acceleration signal (for integration into velocity) would be:

$$\underline{a}_{SF_{Tot}} = \underline{a}_{SF_{Ref}} + \frac{1}{2}\frac{d}{dt}\left(\underline{\alpha} \times \underline{\upsilon}_{Ref}\right) + \frac{1}{2}\left(\underline{\alpha} \times \underline{a}_{SF_{Ref}} + \underline{\upsilon}_{Ref} \times \underline{\omega}\right)$$
$$\underline{\upsilon}_{Ref} \equiv \int_{t_{m-1}}^{t} \underline{a}_{SF_{Ref}}\, d\tau \qquad (8.1.4.1\text{-}10)$$

where

$\underline{a}_{SF_{Tot}}$ = Total specific force acceleration to be integrated into velocity.

$\underline{\alpha}$ = Integral of $\underline{\omega}$ for rotation compensation and sculling terms.

$\underline{\upsilon}_{Ref}$ = $\underline{\upsilon}$ in Equation (7.2.2.2-22) based on the INS navigation reference point specific force acceleration.

t_{m-1} = Time at the end of the previous m computation cycle.

Substituting from (8.1.4.1-9) in (8.1.4.1-10) then finds:

$$\underline{\upsilon}_{Ref} = \int_{t_{m-1}}^{t} G_{Algn}^{-1}\left(\underline{a}_{Accl} - \delta\underline{a}_{Size}\right) d\tau = \underline{\upsilon}^{*} - \delta\underline{\upsilon}_{SizeC}$$
$$\underline{\upsilon}^{*} \equiv G_{Algn}^{-1}\int_{t_{m-1}}^{t} \underline{a}_{Accl}\, d\tau \qquad \delta\underline{\upsilon}_{SizeC} \equiv G_{Algn}^{-1}\int_{t_{m-1}}^{t} \delta\underline{a}_{Size}\, d\tau \qquad (8.1.4.1\text{-}11)$$

$$\begin{aligned}\underline{a}_{SF_{Tot}} &= G_{Algn}^{-1}\left(\underline{a}_{Accl} - \delta\underline{a}_{Size}\right) + \frac{1}{2}\frac{d}{dt}\left(\underline{\alpha} \times \underline{\upsilon}_{Ref}\right) \\ &\quad + \frac{1}{2}\,\underline{\alpha} \times \left[G_{Algn}^{-1}\left(\underline{a}_{Accl} - \delta\underline{a}_{Size}\right)\right] + \frac{1}{2}\left(\underline{\upsilon}^{*} - \delta\underline{\upsilon}_{SizeC}\right) \times \underline{\omega} \\ &= G_{Algn}^{-1}\,\underline{a}_{Accl} - G_{Algn}^{-1}\,\delta\underline{a}_{Size} + \frac{1}{2}\frac{d}{dt}\left(\underline{\alpha} \times \underline{\upsilon}_{Ref}\right) \\ &\quad + \frac{1}{2}\left[\underline{\alpha} \times \left(G_{Algn}^{-1}\,\underline{a}_{Accl}\right) + \underline{\upsilon}^{*} \times \underline{\omega}\right] - \frac{1}{2}\left[\underline{\alpha} \times \left(G_{Algn}^{-1}\,\delta\underline{a}_{Size}\right) + \delta\underline{\upsilon}_{SizeC} \times \underline{\omega}\right]\end{aligned} \qquad (8.1.4.1\text{-}12)$$

where

$\underline{\upsilon}^{*}$ = $\underline{\upsilon}$ in Equation (7.2.2.2-22) based on alignment compensated accelerometer data, but without size effect correction to the INS navigation reference point specific force acceleration.

$\delta\underline{\upsilon}_{SizeC}$ = Size effect correction to $\underline{\upsilon}^{*}$.

The $G_{Algn}^{-1}\,\underline{a}_{Accl}$ term in Equation (8.1.4.1-12) is already accounted for in the Equation (8.1.2.2-4) and (8.1.2.2-6) velocity/position sensor compensation algorithms. Similarly, the

$G_{Algn}^{-1} \, \delta \underline{a}_{Size}$ term in (8.1.4.1-12) is accounted for in (8.1.2.2-4) and (8.1.2.2-6) by the $\delta \underline{\upsilon}_{SizeC_m}$ term which is $\delta \underline{\upsilon}_{SizeC}$ in (8.1.4.1-11) evaluated at time t_m:

$$\delta \underline{\upsilon}_{SizeC_m} \equiv \int_{t_{m-1}}^{t_m} G_{Algn}^{-1} \, \delta \underline{a}_{Size} \, dt \qquad (8.1.4.1\text{-}13)$$

The $\frac{1}{2} \frac{d}{dt} (\underline{\alpha} \times \underline{\upsilon}_{Ref})$ term in Equation (8.1.4.1-12) is handled by the velocity/position rotation compensation algorithms in Equations (7.2.2.2-23), (7.2.2.2.1-7), (7.3.3-9) and (7.3.3.1-16) using the outputs from the (8.1.2.2-4) and (8.1.2.2-6) compensation algorithms. The $\frac{1}{2} \left[\underline{\alpha} \times \left(G_{Algn}^{-1} \, \underline{a}_{Accl} \right) + \underline{\upsilon}^* \times \underline{\omega} \right]$ term in (8.1.4.1-12) is accounted for by the sculling term in Equations (7.2.2.2-23) - (7.2.2.2-24), and in sculling/scrolling compensation algorithms (8.2.2.1-41) and (8.2.3.1-2) to follow later in this chapter. The $\frac{1}{2} \left[\underline{\alpha} \times \left(G_{Algn}^{-1} \, \delta \underline{a}_{Size} \right) + \delta \underline{\upsilon}_{SizeC} \times \underline{\omega} \right]$ term in (8.1.4.1-12) is a correction to the (7.2.2.2-23) sculling term to account for size effect error in the accelerometer data used in this equation. Including the $\frac{1}{2} \left[\underline{\alpha} \times \left(G_{Algn}^{-1} \, \delta \underline{a}_{Size} \right) + \delta \underline{\upsilon}_{SizeC} \times \underline{\omega} \right]$ correction, Equation (7.2.2.2-23) is rewritten as:

$$\Delta \underline{v}_{SF_m}^{B_{I(m-1)}} = \underline{\upsilon}_m + \Delta \underline{v}_{Rot_m} + \Delta \underline{v}_{Scul_m} - \delta \underline{v}_{Scul\text{-}SizeC_m} \qquad (8.1.4.1\text{-}14)$$

with, using (8.1.4.1-11) for $\delta \underline{\upsilon}_{SizeC}$:

$$\delta \underline{v}_{Scul\text{-}SizeC_m} \equiv \int_{t_{m-1}}^{t_m} \frac{1}{2} \left[\underline{\alpha} \times \left(G_{Algn}^{-1} \, \delta \underline{a}_{Size} \right) + \delta \underline{\upsilon}_{SizeC} \times \underline{\omega} \right] dt$$

$$\delta \underline{\upsilon}_{SizeC} \equiv G_{Algn}^{-1} \int_{t_{m-1}}^{t} \delta \underline{a}_{Size} \, d\tau \qquad (8.1.4.1\text{-}15)$$

where

$\delta \underline{v}_{Scul\text{-}SizeC_m}$ = Size effect correction (integrated over an m cycle) applied to the $\Delta \underline{v}_{Scul_m}$ sculling term that was calculated with accelerometer data not containing size effect compensation.

As in Equation (8.1.4.1-12), we also incorporate the $\delta \underline{v}_{Scul\text{-}SizeC_m}$ term into the high resolution Equation (7.3.3-9) B Frame position change increment caused by specific force acceleration, to obtain the enhanced form:

$$\Delta \underline{R}^{B}_{SF_m} = \underline{S}_{\upsilon_m} + \Delta \underline{R}_{Rot_m} + \Delta \underline{R}_{Scrl_m} - \frac{1}{2}\, \delta \underline{v}_{Scul\text{-}SizeC_m}\, T_m \qquad (8.1.4.1\text{-}16)$$

where

T_m = Velocity update time interval.

Equation (8.1.4.1-16) is based on approximating $\delta \underline{v}_{Scul\text{-}SizeC_m}$ as a linear ramping time function over the T_m time interval.

The above development for $\Delta \underline{v}^{B_{I(m-1)}}_{SF_m}$ Equation (8.1.4.1-14) with (8.1.4.1-15) was based on the Equation (7.2.2.2-22) version of the velocity updating increment algorithm. If the alternative (7.2.2.2-6) algorithm version is used, a similar development applies. In this case we start with the equivalent to (7.2.2.2-6):

$$\underline{a}_{SF_{Tot}} = \underline{a}_{SF_{Ref}} + \underline{\alpha} \times \underline{a}_{SF_{Ref}} \qquad (8.1.4.1\text{-}17)$$

and then substitute for $\underline{a}_{SF_{Ref}}$ from (8.1.4.1-9):

$$\begin{aligned} \underline{a}_{SF_{Tot}} &= G^{-1}_{Algn} \left(\underline{a}_{Accl} - \delta \underline{a}_{Size} \right) + \underline{\alpha} \times G^{-1}_{Algn} \left(\underline{a}_{Accl} - \delta \underline{a}_{Size} \right) \\ &= G^{-1}_{Algn}\, \underline{a}_{Accl} - G^{-1}_{Algn}\, \delta \underline{a}_{Size} + \underline{\alpha} \times \left(G^{-1}_{Algn}\, \underline{a}_{Accl} \right) - \underline{\alpha} \times \left(G^{-1}_{Algn}\, \delta \underline{a}_{Size} \right) \end{aligned} \qquad (8.1.4.1\text{-}18)$$

As in (8.1.4.1-12) for the previous development, the $G^{-1}_{Algn}\, \underline{a}_{Accl}$ term in (8.1.4.1-18) is accounted for in the Equation (8.1.2.2-4) and (8.1.2.2-6) velocity/position sensor compensation algorithms, as is the $G^{-1}_{Algn}\, \delta \underline{a}_{Size}$ term using $\delta \underline{\upsilon}_{SizeC_m}$ from (8.1.4.1-13). The $\underline{\alpha} \times \left(G^{-1}_{Algn}\, \delta \underline{a}_{Size} \right)$ term in (8.1.4.1-18) is a correction to the (7.2.2.2-26) composite velocity-rotation-compensation/sculling term to account for size effect error in the accelerometer data used in this equation. Including the $\underline{\alpha} \times \left(G^{-1}_{Algn}\, \delta \underline{a}_{Size} \right)$ correction, Equation (7.2.2.2-26) is rewritten as:

$$\Delta \underline{v}^{B_{I(m-1)}}_{SF_m} = \underline{\upsilon}_m + \Delta \underline{v}_{Rot/Scul_m} - \delta \underline{v}_{Rot/Scul\text{-}SizeC_m} \qquad (8.1.4.1\text{-}19)$$

$$\delta \underline{v}_{Rot/Scul\text{-}SizeC_m} \equiv \int_{t_{m-1}}^{t_m} \underline{\alpha} \times \left(G^{-1}_{Algn}\, \delta \underline{a}_{Size} \right) dt \qquad (8.1.4.1\text{-}20)$$

where

$\delta \underline{v}_{Rot/Scul\text{-}SizeC_m}$ = Size effect correction (integrated over an m cycle) applied to the $\Delta \underline{v}_{Rot/Scul_m}$ composite velocity-rotation-compensation/sculling term that was calculated with accelerometer data not containing size effect compensation.

The subsections to follow develop algorithms for calculating $\delta \underline{\upsilon}_{SizeC_m}$, $\delta \underline{v}_{Scul\text{-}SizeC_m}$ and $\delta \underline{v}_{Rot/Scul\text{-}SizeC_m}$ in Equations (8.1.4.1-14), (8.1.4.1-16) and (8.1.4.1-19) from their analytical definitions in (8.1.4.1-13), (8.1.4.1-15) and (8.1.4.1-20) using $\delta \underline{a}_{Size}$ from (8.1.4.1-9).

8.1.4.1.1 $\delta \underline{\upsilon}_{SizeC_m}$ Size Effect Algorithm

In developing the algorithm for $\delta \underline{\upsilon}_{SizeC_m}$ in (8.1.4.1-13), we take advantage of assumed smallness in the misalignment of the inertial sensor input axes from their nominally assumed orientation along B Frame X, Y and Z axes. This assumption applied to the G_{Algn} accelerometer alignment matrix means that the associated L_{Mis} misalignment matrix is small compared to the identity matrix I. Equation (8.1.1.2-6) rearranged relates G_{Algn} to L_{Mis}:

$$G_{Algn}^{-1} = I - L_{Mis} \tag{8.1.4.1.1-1}$$

Multiplying (8.1.4.1.1-1) by $(I + L_{Mis})$ and dropping L_{Mis}^2 as second order, we see that:

$$G_{Algn}^{-1} (I + L_{Mis}) = (I - L_{Mis})(I + L_{Mis}) = I - L_{Mis}^2 \approx I \tag{8.1.4.1.1-2}$$

or, equivalently:

$$G_{Algn} \approx I + L_{Mis} \tag{8.1.4.1.1-3}$$

Taking the transpose of (8.1.4.1.1-3) finds:

$$G_{Algn}^T = I + L_{Mis}^T \tag{8.1.4.1.1-4}$$

or, by column:

$$\underline{G}_{Algn_k}^T = \underline{u}_k + \underline{L}_{Mis_k}^T \tag{8.1.4.1.1-5}$$

where

$\underline{L}_{Mis_k}^T$ = Vector formed from the k^{th} column of L_{Mis}^T.

We will also make use of the approximate form of (8.1.1.1-8) based on neglecting all but the misalignment error term:

$$\underline{\omega} \approx \underline{\omega}' - K_{Mis}\, \underline{\omega}' \qquad (8.1.4.1.1\text{-}6)$$

where

$\underline{\omega}'$ = Angular rate sensor output compensated for scale factor error but not for misalignment error.

We are now ready to apply (8.1.4.1.1-1), (8.1.4.1.1-5) and (8.1.4.1.1-6) in the (8.1.4.1-13) $\delta\underline{\upsilon}_{SizeC_m}$ equation using (8.1.4.1-9) for $\delta\underline{a}_{Size}$. For the integrand we find after neglecting misalignment products as second order:

$$\begin{aligned} G_{Algn}^{-1}\, \delta\underline{a}_{Size} \approx & \sum_{k=1,3} \left\{ \underline{u}_k \cdot \left[\dot{\underline{\omega}}' \times \underline{l}_k + \underline{\omega}' \times \left(\underline{\omega}' \times \underline{l}_k\right)\right]\right\} \underline{u}_k \\ & - L_{Mis} \sum_{k=1,3} \left\{ \underline{u}_k \cdot \left[\dot{\underline{\omega}}' \times \underline{l}_k + \underline{\omega}' \times \left(\underline{\omega}' \times \underline{l}_k\right)\right]\right\} \underline{u}_k \\ & + \sum_{k=1,3} \left\{ \underline{L}_{Mis_k}^{T} \cdot \left[\dot{\underline{\omega}}' \times \underline{l}_k + \underline{\omega}' \times \left(\underline{\omega}' \times \underline{l}_k\right)\right]\right\} \underline{u}_k \\ - \sum_{k=1,3} & \left\langle \underline{u}_k \cdot \left\{ \left(K_{Mis}\, \dot{\underline{\omega}}'\right) \times \underline{l}_k + \left(K_{Mis}\, \underline{\omega}'\right) \times \left(\underline{\omega}' \times \underline{l}_k\right) + \underline{\omega}' \times \left[\left(K_{Mis}\, \underline{\omega}'\right) \times \underline{l}_k\right] \right\} \right\rangle \underline{u}_k \end{aligned} \qquad (8.1.4.1.1\text{-}7)$$

Let's define some terms to make (8.1.4.1.1-7) at least look a little bit simpler:

$$\delta\underline{a}'_{Size_k} \equiv \dot{\underline{\omega}}' \times \underline{l}_k + \underline{\omega}' \times \left(\underline{\omega}' \times \underline{l}_k\right) \qquad \delta\underline{a}'_{Size} \equiv \sum_{k=1,3} \left(\underline{u}_k \cdot \delta\underline{a}'_{Size_k}\right) \underline{u}_k \qquad (8.1.4.1.1\text{-}8)$$

where

$\delta\underline{a}'_{Size_k}$ = The size effect acceleration vector at the accelerometer k center of seismic mass, but calculated using angular rate sensor data without K_{Mis} misalignment compensation.

$\delta\underline{a}'_{Size}$ = The size effect correction to the accelerometer triad output vector $\underline{a}_{Accl}$, but calculated using accelerometer and angular rate sensor data without L_{Mis} and K_{Mis} misalignment compensation.

Substituting (8.1.4.1.1-8) in (8.1.4.1.1-7) then yields:

$$G_{Algn}^{-1}\, \delta\underline{a}_{Size} = \delta\underline{a}'_{Size} + \Delta\delta\underline{a}_{Size} \qquad (8.1.4.1.1\text{-}9)$$

with

$$\Delta\delta\underline{a}_{Size} \equiv - L_{Mis}\,\delta\underline{a}'_{Size} + \sum_{k=1,3} \left\{ \underline{L}^{T}_{Mis_k} \cdot \delta\underline{a}'_{Size_k} \right\} \underline{u}_k$$
$$- \sum_{k=1,3} \left\langle \underline{u}_k \cdot \left\{ \left(K_{Mis}\,\dot{\underline{\omega}}' \right) \times \underline{l}_k + \left(K_{Mis}\,\underline{\omega}' \right) \times \left(\underline{\omega}' \times \underline{l}_k \right) + \underline{\omega}' \times \left[\left(K_{Mis}\,\underline{\omega}' \right) \times \underline{l}_k \right] \right\} \right\rangle \underline{u}_k \qquad (8.1.4.1.1\text{-}10)$$

where

$\Delta\delta\underline{a}_{Size}$ = Correction to $\delta\underline{a}'_{Size}$ for L_{Mis} and K_{Mis} misalignment.

Using (8.1.4.1.1-9) in (8.1.4.1-13) divides $\delta\underline{\upsilon}_{SizeC_m}$ into two parts:

$$\delta\underline{\upsilon}_{SizeC_m} = \delta\underline{\upsilon}'_{SizeC_m} + \Delta\delta\underline{\upsilon}_{SizeC_m} \qquad (8.1.4.1.1\text{-}11)$$

$$\delta\underline{\upsilon}'_{SizeC_m} \equiv \int_{t_{m-1}}^{t_m} \delta\underline{a}'_{Size}\, dt \qquad \Delta\delta\underline{\upsilon}_{SizeC_m} \equiv \int_{t_{m-1}}^{t_m} \Delta\delta\underline{a}_{Size}\, dt \qquad (8.1.4.1.1\text{-}12)$$

with $\delta\underline{a}'_{Size}$ and $\Delta\delta\underline{a}_{Size}$ as defined in (8.1.4.1.1-8) and (8.1.4.1.1-10) and where

$\delta\underline{\upsilon}'_{SizeC_m}$ = The size effect correction to the integrated accelerometer triad output increment vector $\underline{\upsilon}_m$, but calculated using accelerometer and angular rate sensor data without L_{Mis} and K_{Mis} misalignment compensation.

$\Delta\delta\underline{\upsilon}_{SizeC_m}$ = Correction to $\delta\underline{\upsilon}'_{SizeC_m}$ for L_{Mis} and K_{Mis} misalignment.

The $\delta\underline{\upsilon}'_{SizeC_m}$ equation in (8.1.4.1.1-12) can be viewed as an approximation to the correct $\delta\underline{\upsilon}_{SizeC_m}$ value based on the assumption that sensor misalignments are small, hence, can be ignored in the $\delta\underline{\upsilon}_{SizeC_m}$ determination. For most applications, the error introduced by this approximation is negligibly small. The $\Delta\delta\underline{\upsilon}_{SizeC_m}$ correction term can be applied for those unusual applications when sensor misalignment effects must be included in the size effect calculations. The following subsections find algorithms for $\delta\underline{\upsilon}'_{SizeC_m}$ and $\Delta\delta\underline{\upsilon}_{SizeC_m}$ in (8.1.4.1.1-11) and (8.1.4.1.1-12).

8.1.4.1.1.1 $\delta\underline{\upsilon}'_{SizeC_m}$ Size Effect Term Algorithm

The algorithm for evaluating $\delta\underline{\upsilon}'_{SizeC_m}$ as defined by Equation (8.1.4.1.1-12) is determined by first expanding Equations (8.1.4.1.1-8) for $\delta\underline{a}'_{Size}$ in component form, using the following for the individual vector constituents:

$$\underline{\omega}' = \left(\omega'_X, \omega'_Y, \omega'_Z\right)^T \qquad \underline{\dot{\omega}}' = \left(\dot{\omega}'_X, \dot{\omega}'_Y, \dot{\omega}'_Z\right)^T \qquad \underline{l}_k = \left(l_{X_k}, l_{Y_k}, l_{Z_k}\right)^T \qquad (8.1.4.1.1.1\text{-}1)$$

where

$\omega'_X, \omega'_Y, \omega'_Z$ = B Frame X, Y, Z components of $\underline{\omega}'$.

$l_{X_k}, l_{Y_k}, l_{Z_k}$ = B Frame X, Y, Z components of $\underline{l}_k$.

Substituting (8.1.4.1.1.1-1) into Equations (8.1.4.1.1-8) obtains for the Y component of $\delta\underline{a}'_{Size}$ (representative):

$$\delta a'_{Size_Y} = - l_{Z_2}\,\dot{\omega}'_X + l_{X_2}\,\dot{\omega}'_Z + l_{Z_2}\,\omega'_Y\,\omega'_Z + l_{X_2}\,\omega'_Y\,\omega'_X - l_{Y_2}\left(\omega'^2_Z + \omega'^2_X\right) \qquad (8.1.4.1.1.1\text{-}2)$$

where

$\delta a'_{Size_Y}$ = B Frame Y axis component of $\delta\underline{a}'_{Size}$.

The computer algorithm for integration of Equation (8.1.4.1.1.1-2) into the Equation (8.1.4.1.1-12) $\delta\underline{\upsilon}'_{SizeC_m}$ size effect term is developed by analyzing the integral of the two characteristic terms in (8.1.4.1.1.1-2) over a transformation cycle:

$$\gamma_{i_m} \equiv \int_{t_{m-1}}^{t_m} \dot{\omega}'_i\,dt \qquad \gamma_{ij_m} \equiv \int_{t_{m-1}}^{t_m} \omega'_i\,\omega'_j\,dt \qquad (8.1.4.1.1.1\text{-}3)$$

where

m = Acceleration-transformation/velocity-update cycle index.

$\gamma_{i_m}, \gamma_{ij_m}$ = Integrated characteristic terms in (8.1.4.1.1.1-2) over an m cycle.

i = B Frame angular rate input axis (X, Y, or Z).

The γ_{i_m} term in (8.1.4.1.1.1-3) is by direct integration:

$$\gamma_{i_m} = \omega'_{i_m} - \omega'_{i_{m-1}} \qquad (8.1.4.1.1.1\text{-}4)$$

where

$\omega'_{i_m}, \omega'_{i_{m-1}}$ = ω'_i at the current m cycle time t_m and the previous m cycle time t_{m-1}.

A digital algorithm for (8.1.4.1.1.1-4) is derived from the following approximations for ω'_{i_m} and $\omega'_{i_{m-1}}$:

$$\omega'_{i_m} \approx \frac{1}{2} f_{Size}\left(\Delta\alpha'_{i_{m+}} + \Delta\alpha'_{i_m}\right) \qquad \omega'_{i_{m-1}} \approx \frac{1}{2} f_{Size}\left(\Delta\alpha'_{i_{(m-1)+}} + \Delta\alpha'_{i_{m-1}}\right) \tag{8.1.4.1.1.1-5}$$

in which

$$\Delta\alpha'_{i_l} \equiv \int_{t_{l-1}}^{t_l} d\alpha'_i \tag{8.1.4.1.1.1-6}$$

where

l = Size effect algorithm computation rate index (typically equal to the sculling algorithm iteration rate).

$d\alpha'_i = \omega'_i dt$ = Differential integrated i axis scaled angular rate increment (i.e., analytical representation of scaled pulse output from the i axis strapdown angular rate sensor).

$\Delta\alpha'_{i_l}$ = Integrated B Frame i axis inertial angular rotation rate from t_{l-1} to t_l (with scale factor calibration but without misalignment correction).

f_{Size} = Size effect algorithm computation frequency equal to the reciprocal of the l cycle time period t_l - t_{l-1}.

$\Delta\alpha'_{i_m}$, $\Delta\alpha'_{i_{m-1}}$ = $\Delta\alpha'_{i_l}$ at the current and previous m cycle (at $t_l = t_m$ and $t_l = t_{m-1}$).

$\Delta\alpha'_{i_{m+}}$, $\Delta\alpha'_{i_{(m-1)+}}$ = $\Delta\alpha'_{i_l}$ immediately following $\Delta\alpha'_{i_m}$, $\Delta\alpha'_{i_{m-1}}$ (i.e., for t_l immediately following t_m and t_{m-1}).

Substituting (8.1.4.1.1.1-5) in (8.1.4.1.1.1-4) then gives for γ_{i_m}:

$$\gamma_{i_m} = \frac{1}{2} f_{Size}\left(\Delta\alpha'_{i_{m+}} + \Delta\alpha'_{i_m} - \Delta\alpha'_{i_{(m-1)+}} - \Delta\alpha'_{i_{m-1}}\right) \tag{8.1.4.1.1.1-7}$$

In practice, to implement (8.1.4.1.1.1-7) as shown, the m cycle transformation computation would require a delay of one size effect calculation cycle to measure $\Delta\alpha'_{i_{m+}}$. To avoid the time delay, the following (8.1.4.1.1.1-8) approximation to Equation (8.1.4.1.1.1-7) can be used for γ_{i_m}. The approximation is equivalent to a phase shift in γ_{i_m} of half a size effect computation cycle. For constant angular acceleration, the (8.1.4.1.1.1-8) algorithm yields the identical results as (8.1.4.1.1.1-7).

$$\gamma_{i_m} \approx f_{Size}\left(\Delta\alpha'_{i_m} - \Delta\alpha'_{i_{m-1}}\right) \tag{8.1.4.1.1.1-8}$$

An important property of the (8.1.4.1.1.1-8) approximation is that its sum over successive m cycles equals the current $\Delta\alpha'_{i_m}$ minus the original $\Delta\alpha'_{i_m}$ at the start of the summation. This characteristic is also present in the (8.1.4.1.1.1-7) form as well as the fundamental Equation (8.1.4.1.1.1-4) form which the algorithm has been designed to emulate.

A construct of a digital integration algorithm for γ_{ij_m} in Equation (8.1.4.1.1.1-3) is prepared by first introducing an intermediate parameter:

$$\eta_{ij_m} \equiv \frac{1}{f_{Size}} \int_{t_{m-1}}^{t_m} \omega'_i \, \omega'_j \, dt \qquad (8.1.4.1.1.1\text{-}9)$$

with which, from (8.1.4.1.1.1-3):

$$\gamma_{ij_m} = f_{Size} \eta_{ij_m} \qquad (8.1.4.1.1.1\text{-}10)$$

where

η_{ij_m} = Frequency normalized γ_{ij_m} parameter.

As will soon be apparent, introducing the intermediate η_{ij_m} parameter has certain algorithmic computational simplifications. Then from (8.1.4.1.1.1-9), as in Equations (7.1.1.1.1-4), we write:

$$\Delta\eta_{ij_l} \equiv \frac{1}{f_{Size}} \int_{t_{l-1}}^{t_l} \omega'_i \, \omega'_j \, dt \qquad \eta_{ij_l} = \eta_{ij_{l-1}} + \Delta\eta_{ij_l}$$
$$\eta_{ij_m} = \eta_{ij_l}(t_l = t_m) \qquad \eta_{ij_l} = 0 \quad \text{At } t = t_{m-1} \qquad (8.1.4.1.1.1\text{-}11)$$

where

$\Delta\eta_{ij_l}$ = Integrated frequency normalized characteristic term in (8.1.4.1.1.1-2) over an l cycle.

Evaluation of the Equation (8.1.4.1.1.1-11) $\Delta\eta_{ij_l}$ integral over an l cycle is based on approximating ω'_i and ω'_j by their average values over the l cycle:

$$\omega'_i \approx f_{Size} \Delta\alpha'_{i_l} \qquad \omega'_j \approx f_{Size} \Delta\alpha'_{j_l} \qquad (8.1.4.1.1.1\text{-}12)$$

Substituting (8.1.4.1.1.1-12) into the (8.1.4.1.1.1-11) $\Delta\eta_{ij_l}$ expression yields the approximate form:

$$\Delta\eta_{ij_l} \approx \Delta\alpha'_{i_l} \, \Delta\alpha'_{j_l} \qquad (8.1.4.1.1.1\text{-}13)$$

The overall $\delta\underline{\upsilon}'_{SizeC_m}$ size effect correction algorithm is obtained from Equation (8.1.4.1.1-12) using (8.1.4.1.1.1-2) as the representative Y axis integrand component, with the individual term integral forms provided by Equation (8.1.4.1.1.1-8) for γ_{i_m} and (8.1.4.1.1.1-10) for γ_{ij_m}, using η_{ij_m} from (8.1.4.1.1.1-11) with (8.1.4.1.1.1-13) for $\Delta\eta_{ij_l}$. The final combined result is:

$$\Delta\alpha'_{i_l} \equiv \int_{t_{l-1}}^{t_l} d\alpha'_i \qquad \text{Summation Of Integrated Scaled Angular Rate Output Increments From Angular Rate Sensors}$$

$$\eta_{ij_l} = \eta_{ij_{l-1}} + \Delta\alpha'_{i_l}\,\Delta\alpha'_{j_l}$$

$$\eta_{ij_m} = \eta_{ij_l}(t_l = t_m) \qquad \eta_{ij_l} = 0 \quad \text{At } t = t_{m-1} \tag{8.1.4.1.1.1-14}$$

$$\Delta\alpha'_{i_m} = \Delta\alpha'_{i_l} \quad \text{At } t = t_m$$

$$\Delta\alpha'_{i_{(l\,:\,m)-1}} = \Delta\alpha'_{i_l} \quad \text{Immediately preceding } t = t_m.$$

$$\delta\upsilon'_{SizeCY_m} = f_{Size}\Big[-l_{Z_2}\left(\Delta\alpha'_{X_m} - \Delta\alpha'_{X_{m-1}}\right) + l_{X_2}\left(\Delta\alpha'_{Z_m} - \Delta\alpha'_{Z_{m-1}}\right) + l_{Z_2}\,\eta_{YZ_m} + l_{X_2}\,\eta_{XY_m} - l_{Y_2}\left(\eta_{ZZ_m} + \eta_{XX_m}\right)\Big] \tag{8.1.4.1.1.1-15}$$

$$\delta\upsilon'_{SizeCZ_m},\ \delta\upsilon'_{SizeCX_m} = \text{Similarly by permuting subscripts.}$$

where

$$\delta\upsilon'_{SizeCX_m},\ \delta\upsilon'_{SizeCY_m},\ \delta\upsilon'_{SizeCZ_m} = \text{X, Y, Z Components of } \delta\underline{\upsilon}'_{SizeC_m}.$$

The $\Delta\alpha'_{i_{(l\,:\,m)-1}}$ representation in (8.1.4.1.1.1-14) has been provided for the Equation (8.1.3.3-5) quantization expression which will be utilized in Section 8.1.4.1.4 to compensate $\Delta\alpha_i$ for sensor error.

8.1.4.1.1.2 $\Delta\delta\underline{\upsilon}_{SizeC_m}$ Size Effect Correction Algorithm

The $\Delta\delta\underline{\upsilon}_{SizeC_m}$ integrand in (8.1.4.1.1-12) will be evaluated using the following for the misalignment matrices based on the approximation that L_{Mis} and K_{Mis} are small, hence, their diagonal terms are second order and negligible (See Equations (8.1.1.1-6), (8.1.1.1.1-16), (8.1.1.2-6) and (8.1.1.2.1-14)):

$$K_{Mis} \approx \begin{bmatrix} 0 & K_{Mis_{XY}} & K_{Mis_{XZ}} \\ K_{Mis_{YX}} & 0 & K_{Mis_{YZ}} \\ K_{Mis_{ZX}} & K_{Mis_{ZY}} & 0 \end{bmatrix} \qquad L_{Mis} \approx \begin{bmatrix} 0 & L_{Mis_{XY}} & L_{Mis_{XZ}} \\ L_{Mis_{YX}} & 0 & L_{Mis_{YZ}} \\ L_{Mis_{ZX}} & L_{Mis_{ZY}} & 0 \end{bmatrix} \tag{8.1.4.1.1.2-1}$$

where

$K_{Mis_{ij}}, L_{Mis_{ij}}$ = Elements in row i, column j of K_{Mis} and L_{Mis}.

Using (8.1.4.1.1-10) for $\Delta\delta\underline{a}_{Size}$, (8.1.4.1.1-8) for $\delta\underline{a}'_{Size}$ and $\delta\underline{a}'_{Size_k}$, (8.1.4.1.1.1-1) for the vector components, and (8.1.4.1.1.2-1) for the misalignment matrices, the $\Delta\delta\underline{\upsilon}_{SizeC_m}$ integrand in (8.1.4.1.1-12) becomes for the Y-axis component (as an example):

$$\begin{aligned} \Delta\delta a_{Size_Y} = {} & - L_{Mis_{YX}} \left(\delta a'_{Size_X} - \delta a'_{Size/X_2}\right) - L_{Mis_{YZ}} \left(\delta a'_{Size_Z} - \delta a'_{Size/Z_2}\right) \\ & - K_{Mis_{ZX}}\, l_{X_2}\, \dot{\omega}'_X - \left(K_{Mis_{ZY}}\, l_{X_2} - K_{Mis_{XY}}\, l_{Z_2}\right) \dot{\omega}'_Y + K_{Mis_{XZ}}\, l_{Z_2}\, \dot{\omega}'_Z \\ & - \left(K_{Mis_{YX}}\, l_{X_2}\right) {\omega'_X}^2 - \left(K_{Mis_{XY}}\, l_{X_2} + K_{Mis_{ZY}} l_{Z_2}\right) {\omega'_Y}^2 - \left(K_{Mis_{YZ}}\, l_{Z_2}\right) {\omega'_Z}^2 \\ & + \left(2\, K_{Mis_{XY}} l_{Y_2} - K_{Mis_{ZX}} l_{Z_2}\right) \omega'_X\, \omega'_Y + \left(2\, K_{Mis_{ZY}} l_{Y_2} - K_{Mis_{XZ}} l_{X_2}\right) \omega'_Y\, \omega'_Z \\ & + \left[2 \left(K_{Mis_{XZ}} + K_{Mis_{ZX}}\right) l_{Y_2} - K_{Mis_{YZ}}\, l_{X_2} - K_{Mis_{YX}}\, l_{Z_2}\right] \omega'_Z\, \omega'_X \end{aligned} \tag{8.1.4.1.1.2-2}$$

$$\delta a'_{Size/X_k} = - l_{Y_k}\, \dot{\omega}'_Z + l_{Z_k}\, \dot{\omega}'_Y + l_{Y_k}\, \omega'_X\, \omega'_Y + l_{Z_k}\, \omega'_Z\, \omega'_X - l_{X_k} \left({\omega'_Y}^2 + {\omega'_Z}^2\right)$$

$\delta a'_{Size/Z_k}$ = Similarly by permuting subscripts.

where

$\delta a'_{Size_X}, \delta a'_{Size_Z}$ = B Frame X and Y axis components of $\delta\underline{a}'_{Size}$.

$\delta a'_{Size/X_k}, \delta a'_{Size/Z_k}$ = B Frame X and Y axis components of $\delta\underline{a}'_{Size_k}$ (k = 1, 2 or 3 for the X, Y or Z axis accelerometer).

Note from the analytical definitions of $\delta\underline{a}'_{Size}$ and $\delta\underline{a}'_{Size_k}$ in (8.1.4.1.1-8), that $\delta a'_{Size_Y}$ equals $\delta a'_{Size/Y_2}$, and similarly for the X and Z components (i.e., $\delta a'_{Size_X} = \delta a'_{Size/X_1}$ and $\delta a'_{Size_Z} = \delta a'_{Size/Z_3}$).

Integrating (8.1.4.1.1.2-2) from t_{m-1} to t_m, using (8.1.4.1.1.1-9) for the $\omega'_i\, \omega'_j$ integrals, (8.1.4.1.1.1-3) and (8.1.4.1.1.1-8) for the $\dot{\omega}'_i$ integrals, and (8.1.4.1.1-12) for the $\delta a'_{Size_i}$ integrals, we then find for the $\Delta\delta\underline{\upsilon}_{SizeC_m}$ algorithm:

$\eta_{ij_m}, \Delta\alpha'_{j_m}$ = From Algorithm Equations (8.1.4.1.1.1-14).

$\delta\upsilon'_{SizeCi_m}$ = From Algorithm Equations (8.1.4.1.1.1-15).

$$\begin{aligned}
\Delta\delta\upsilon_{SizeCY_m} = & - L_{Mis_{YX}}\left(\delta\upsilon'_{SizeCX_m} - \delta\upsilon'_{SizeCX/2_m}\right) - L_{Mis_{YZ}}\left(\delta\upsilon'_{SizeCZ_m} - \delta\upsilon'_{SizeCZ/2_m}\right) \\
& + f_{Size}\Big\{- K_{Mis_{ZX}}\, l_{X_2}\left(\Delta\alpha'_{X_m} - \Delta\alpha'_{X_{m-1}}\right) - \left(K_{Mis_{ZY}}\, l_{X_2} - K_{Mis_{XY}}\, l_{Z_2}\right)\left(\Delta\alpha'_{Y_m} - \Delta\alpha'_{Y_{m-1}}\right) \\
& \qquad + K_{Mis_{XZ}}\, l_{Z_2}\left(\Delta\alpha'_{Z_m} - \Delta\alpha'_{Z_{m-1}}\right) \\
& - \left(K_{Mis_{YX}}\, l_{X_2}\right)\eta_{XX_m} - \left(K_{Mis_{XY}}\, l_{X_2} + K_{Mis_{ZY}}\, l_{Z_2}\right)\eta_{YY_m} - \left(K_{Mis_{YZ}}\, l_{Z_2}\right)\eta_{ZZ_m} \\
& + \left(2\, K_{Mis_{XY}}\, l_{Y_2} - K_{Mis_{ZX}}\, l_{Z_2}\right)\eta_{XY_m} + \left(2\, K_{Mis_{ZY}}\, l_{Y_2} - K_{Mis_{XZ}}\, l_{X_2}\right)\eta_{YZ_m} \\
& + \left[2\left(K_{Mis_{XZ}} + K_{Mis_{ZX}}\right) l_{Y_2} - K_{Mis_{YZ}}\, l_{X_2} - K_{Mis_{YX}}\, l_{Z_2}\right]\eta_{ZX_m}\Big\}
\end{aligned} \tag{8.1.4.1.1.2-3}$$

$\Delta\delta\upsilon_{SizeCZ_m}, \Delta\delta\upsilon_{SizeCX_m}$ = Similarly by permuting subscripts.

with the newly defined vector:

$$\delta\underline{\upsilon}'_{SizeC/k_m} \equiv \int_{t_{m-1}}^{t_m} \delta\underline{a}'_{Size_k}\, dt \tag{8.1.4.1.1.2-4}$$

and its components using (8.1.4.1.1-8) for $\delta\underline{a}'_{Size_k}$:

$$\begin{aligned}
\delta\upsilon'_{SizeCX/k_m} = f_{Size}\Big[& - l_{Y_k}\left(\Delta\alpha'_{Z_m} - \Delta\alpha'_{Z_{m-1}}\right) + l_{Z_k}\left(\Delta\alpha'_{Y_m} - \Delta\alpha'_{Y_{m-1}}\right) \\
& + l_{Y_k}\,\eta_{XY_m} + l_{Z_k}\,\eta_{ZX_m} - l_{X_k}\left(\eta_{YY_m} + \eta_{ZZ_m}\right)\Big]
\end{aligned} \tag{8.1.4.1.1.2-5}$$

$\delta\upsilon'_{SizeCY/k_m}, \delta\upsilon'_{SizeCZ/k_m}$ = Similarly by permuting subscripts.

where

$\delta\Delta\upsilon_{SizeCX_m}, \delta\Delta\upsilon_{SizeCY_m}, \delta\Delta\upsilon_{SizeCZ_m}$ = B Frame X, Y, Z components of $\Delta\delta\underline{\upsilon}_{SizeC_m}$ defined in Equation (8.1.4.1.1-12).

$\delta\underline{\upsilon}'_{SizeC/k_m}$ = Integral of $\delta\underline{a}'_{Size_k}$ from t_{m-1} to t_m.

$\delta\upsilon'_{SizeCX/k_m}, \delta\upsilon'_{SizeCY/k_m}, \delta\upsilon'_{SizeCZ/k_m}$ = B Frame X, Y, Z components of $\delta\underline{\upsilon}'_{SizeC/k_m}$.

8.1.4.1.2 $\delta\underline{v}_{Scul\text{-}SizeC_m}$ Sculling Size Effect Algorithm

The algorithm for $\delta\underline{v}_{Scul\text{-}SizeC_m}$ is based on an approximate form of (8.1.4.1-15) that ignores sensor misalignment effects as second order, and eliminates other terms based on the relative frequency and magnitude of angular vibration compared to other angular rate effects. The algorithm derivation begins with approximating the G_{Algn} matrix in (8.1.4.1-15) by identity and $\underline{G}^T_{Algn\,k}$ as $\underline{u}_k$ (see (8.1.4.1.1-5)) so that (8.1.4.1-15) with (8.1.4.1-9) for $\delta\underline{a}_{Size}$ becomes:

$$\delta\underline{v}_{Scul\text{-}SizeC_m} \approx \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha} \times \delta\underline{a}_{Size} + \delta\underline{\upsilon}_{SizeC} \times \underline{\omega}\right) dt$$

$$\delta\underline{a}_{Size} \approx \sum_{k=1,3} \left\{ \underline{u}_k \cdot \left[\dot{\underline{\omega}} \times \underline{l}_k + \underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k\right)\right]\right\} \underline{u}_k \qquad (8.1.4.1.2\text{-}1)$$

$$\delta\underline{\upsilon}_{SizeC} \approx \int_{t_{m-1}}^{t} \delta\underline{a}_{Size}\, d\tau \approx \sum_{k=1,3} \left\{ \underline{u}_k \cdot \int_{t_{m-1}}^{t} \left[\dot{\underline{\omega}} \times \underline{l}_k + \underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k\right)\right] d\tau \right\} \underline{u}_k$$

We then define the angular rate $\underline{\omega}$ as the sum of vibration and remaining lower frequency components such that:

$$\underline{\omega} = \underline{\omega}_{Lo\text{-}f} + \underline{\omega}_{Vib}$$

$$\dot{\underline{\omega}} = \dot{\underline{\omega}}_{Lo\text{-}f} + \dot{\underline{\omega}}_{Vib}$$

$$\underline{\alpha} = \int_{t_{m-1}}^{t} \underline{\omega}\, d\tau = \underline{\alpha}_{Lo\text{-}f} + \underline{\alpha}_{Vib} \qquad (8.1.4.1.2\text{-}2)$$

$$\underline{\alpha}_{Lo\text{-}f} \equiv \int_{t_{m-1}}^{t} \underline{\omega}_{Lo\text{-}f}\, d\tau \qquad \underline{\alpha}_{Vib} \equiv \int_{t_{m-1}}^{t} \underline{\omega}_{Vib}\, d\tau$$

where

$\underline{\omega}_{Vib}$ = Higher frequency components of $\underline{\omega}$ attributed to angular vibration.

$\underline{\omega}_{Lo\text{-}f}$ = Remaining lower frequency components of $\underline{\omega}$.

$\underline{\alpha}_{Lo\text{-}f}$, $\underline{\alpha}_{Vib}$ = Integrals of $\underline{\omega}_{Lo\text{-}f}$ and $\underline{\omega}_{Vib}$ since time t_{m-1}.

To assess the relative magnitude of terms in (8.1.4.1.2-1), we substitute (8.1.4.1.2-2) in (8.1.4.1.2-1), expand the $\delta\underline{v}_{Scul\text{-}SizeC_m}$ integrand in terms of the new $\underline{\omega}_{Vib}$, $\underline{\omega}_{Lo\text{-}f}$, $\dot{\underline{\omega}}_{Lo\text{-}f}$, $\dot{\underline{\omega}}_{Vib}$, $\underline{\alpha}_{Lo\text{-}f}$ and $\underline{\alpha}_{Vib}$ parameters, and assess the order of magnitude of each of the resulting components in the expanded $\delta\underline{v}_{Scul\text{-}SizeC_m}$ integrand form. The result for each of the $\delta\underline{v}_{Scul\text{-}SizeC_m}$ integrand constituents is:

$$
\begin{aligned}
&\underline{\omega}_{Lo\text{-}f} \overset{o}{=} \omega_{Lo\text{-}f} \\
&\underline{\alpha}_{Lo\text{-}f} \overset{o}{=} \omega_{Lo\text{-}f}\, T_m \\
&\underline{\alpha}_{Vib} \overset{o}{=} \theta_{Vib} \sin\left(\Omega_{Vib}\, t\right) \overset{o}{=} \theta_{Vib} \\
&\underline{\omega}_{Vib} = \dot{\underline{\alpha}}_{Vib} \overset{o}{=} \theta_{Vib}\, \Omega_{Vib} \cos\left(\Omega_{Vib}\, t\right) \overset{o}{=} \theta_{Vib}\, \Omega_{Vib} \\
&\dot{\underline{\omega}}_{Vib} \overset{o}{=} \theta_{Vib}\, \Omega_{Vib}^2 \\
&\dot{\underline{\omega}}_{Vib} \times \underline{l}_k \overset{o}{=} \theta_{Vib}\, \Omega_{Vib}^2\, l \\
&\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right) \overset{o}{=} \theta_{Vib}^2\, \Omega_{Vib}^2\, l \\
&\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right) \overset{o}{=} \theta_{Vib}\, \omega_{Lo\text{-}f}\, \Omega_{Vib}\, l \\
&\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right) \overset{o}{=} \theta_{Vib}\, \omega_{Lo\text{-}f}\, \Omega_{Vib}\, l \\
&\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right) \overset{o}{=} \omega_{Lo\text{-}f}^2\, l \\
&\int_{t_{m-1}}^{t} \left[\dot{\underline{\omega}}_{Vib} \times \underline{l}_k\right] d\tau \overset{o}{=} \theta_{Vib}\, \Omega_{Vib}\, l \\
&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} \theta_{Vib}^2\, \Omega_{Vib}\, l \\
&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} \theta_{Vib}\, \omega_{Lo\text{-}f}\, l \\
&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} \theta_{Vib}\, \omega_{Lo\text{-}f}\, l \\
&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} \omega_{Lo\text{-}f}^2\, T_m\, l
\end{aligned}
\qquad (8.1.4.1.2\text{-}3)
$$

where

$\overset{o}{=}$ = Notation signifying "is of the order of magnitude of".

θ_{Vib} = Representative angle vibration amplitude.

Ω_{Vib} = Representative angular vibration frequency.

$\omega_{Lo\text{-}f}$ = Representative $\underline{\omega}_{Lo\text{-}f}$ magnitude.

l = Representative $\underline{l}_k$ magnitude.

T_m = Time interval from t_{m-1} to t_m for the computer m cycle update rate.

To make numerical comparisons between the (8.1.4.1.2-3) terms, we will use the following as representative of moderate to severe environment values: $\omega_{Lo\text{-}f} = 300$ deg/sec ≈ 6 rad/sec and from Section 7.4, $\theta_{Vib} = 0.4$ milli-rad and $\Omega_{Vib} = 2\,\pi \times 50$ Hz ≈ 300 rad/sec. Based on a representative T_m update cycle time value of 0.01 sec, Equations (8.1.4.1.2-3) become:

$$
\begin{aligned}
&\underline{\omega}_{Lo\text{-}f} \overset{o}{=} 6 \qquad &&\int_{t_{m-1}}^{t} \left[\dot{\underline{\omega}}_{Vib} \times \underline{l}_k\right] d\tau \overset{o}{=} 0.12\, l \\
&\underline{\alpha}_{Lo\text{-}f} \overset{o}{=} 0.06 \\
&\underline{\alpha}_{Vib} \overset{o}{=} 0.0004 &&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} 0.000048\, l \\
&\underline{\omega}_{Vib} \overset{o}{=} 0.12 \\
&\dot{\underline{\omega}}_{Vib} \overset{o}{=} 36 &&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} 0.0024\, l \\
&\dot{\underline{\omega}}_{Vib} \times \underline{l}_k \overset{o}{=} 36\, l \\
&\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right) \overset{o}{=} 0.0144\, l &&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} 0.0024\, l \\
&\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right) \overset{o}{=} 0.72\, l \\
&\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right) \overset{o}{=} 0.72\, l &&\int_{t_{m-1}}^{t} \left[\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)\right] d\tau \overset{o}{=} 0.36\, l \\
&\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right) \overset{o}{=} 36\, l
\end{aligned}
\tag{8.1.4.1.2-4}
$$

Based on (8.1.4.1.2-4), we can ignore the $\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)$, $\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)$ and $\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)$ terms in the (8.1.4.1.2-1) $\delta\underline{a}_{Size}$ expression compared to the $\dot{\underline{\omega}}_{Vib} \times \underline{l}_k$ and $\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)$ terms, and we can ignore $\underline{\omega}_{Vib}$ compared to $\underline{\omega}_{Lo\text{-}f}$. Similarly, we can also ignore the integrated $\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)$, $\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Vib} \times \underline{l}_k\right)$ and $\underline{\omega}_{Vib} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)$ terms in the (8.1.4.1.2-1) $\delta\underline{\upsilon}_{SizeC}$ expression compared to the integrated $\dot{\underline{\omega}}_{Vib} \times \underline{l}_k$ and $\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k\right)$ terms, and we can ignore $\underline{\alpha}_{Vib}$ compared to $\underline{\alpha}_{Lo\text{-}f}$. Based on these findings, we define the following for the potentially significant $\delta\underline{a}_{Size}$ and $\delta\underline{\upsilon}_{SizeC}$ constituents:

$$\delta\underline{a}''_{Size} \equiv \sum_{k=1,3} \left[\underline{u}_k \cdot \left(\dot{\underline{\omega}} \times \underline{l}_k \right) \right] \underline{u}_k \qquad \delta\underline{\upsilon}''_{SizeC} \equiv \int_{t_{m-1}}^{t} \delta\underline{a}''_{Size} \, d\tau$$

$$\delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} \equiv \sum_{k=1,3} \left\{ \underline{u}_k \cdot \left[\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k \right) \right] \right\} \underline{u}_k \qquad (8.1.4.1.2\text{-}5)$$

$$\delta\underline{\upsilon}_{SizeC_{Lo\text{-}f/\omega^2}} \equiv \int_{t_{m-1}}^{t} \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} \, d\tau$$

where

$\delta\underline{a}''_{Size}$ = Approximation for $\delta\underline{a}_{Size}$ that ignores all $\underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k \right)$ terms.

$\delta\underline{\upsilon}''_{SizeC}$ = The integral of $\delta\underline{a}''_{Size}$ from time t_{m-1}.

$\delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}}$ = The $\underline{\omega}_{Lo\text{-}f} \times \left(\underline{\omega}_{Lo\text{-}f} \times \underline{l}_k \right)$ portion of $\underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k \right)$.

$\delta\underline{\upsilon}_{SizeC_{Lo\text{-}f/\omega^2}}$ = The integral of $\delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}}$ from time t_{m-1}.

Using the (8.1.4.1.2-5) definitions and the previous relative magnitude assumptions, the (8.1.4.1.2-1) $\delta\underline{v}_{Scul\text{-}SizeC_m}$ integrand with (8.1.4.1.2-2) becomes:

$$\begin{aligned} \underline{\alpha} \times \left(\delta\underline{a}_{Size} \right) + \delta\underline{\upsilon}_{SizeC} \times \underline{\omega} \; &\approx \; \underline{\alpha} \times \delta\underline{a}''_{Size} + \delta\underline{\upsilon}''_{SizeC} \times \underline{\omega} \\ &\quad + \left(\underline{\alpha}_{Lo\text{-}f} + \underline{\alpha}_{Vib} \right) \times \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} + \delta\underline{\upsilon}_{SizeC_{Lo\text{-}f/\omega^2}} \times \left(\underline{\omega}_{Lo\text{-}f} + \underline{\omega}_{Vib} \right) \qquad (8.1.4.1.2\text{-}6) \\ &\approx \; \underline{\alpha} \times \delta\underline{a}''_{Size} + \delta\underline{\upsilon}''_{SizeC} \times \underline{\omega} + \underline{\alpha}_{Lo\text{-}f} \times \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} + \delta\underline{\upsilon}_{SizeC_{Lo\text{-}f/\omega^2}} \times \underline{\omega}_{Lo\text{-}f} \end{aligned}$$

Based on the definition for the Lo-f terms in (8.1.4.1.2-6) as being composed of low frequency components, we can approximate these terms as constant over an m update cycle. Then the following shows that the last two terms in (8.1.4.1.2-6) sum to approximately zero.

$$\begin{aligned} &\underline{\alpha}_{Lo\text{-}f} \approx \underline{\omega}_{Lo\text{-}f} \left(t - t_{m-1} \right) \\ &\delta\underline{\upsilon}_{SizeC_{Lo\text{-}f/\omega^2}} \approx \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} \left(t - t_{m-1} \right) \\ &\underline{\alpha}_{Lo\text{-}f} \times \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} + \delta\underline{\upsilon}_{SizeC_{Lo\text{-}f/\omega^2}} \times \underline{\omega}_{Lo\text{-}f} \\ &\qquad \approx \left(\underline{\omega}_{Lo\text{-}f} \times \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} \right) \left(t - t_{m-1} \right) + \left(\delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}} \times \underline{\omega}_{Lo\text{-}f} \right) \left(t - t_{m-1} \right) = 0 \end{aligned} \qquad (8.1.4.1.2\text{-}7)$$

With (8.1.4.1.2-7), we see from (8.1.4.1.2-6) that $\underline{\alpha} \times \left(\delta\underline{a}_{Size} \right) + \delta\underline{\upsilon}_{SizeC} \times \underline{\omega}$ can be approximated as $\underline{\alpha} \times \delta\underline{a}''_{Size} + \delta\underline{\upsilon}''_{SizeC} \times \underline{\omega}$. It is also useful to note that this approximation is valid under zero vibration for any $\underline{\omega}_{Lo\text{-}f}$ value, not only the value used in (8.1.4.1.2-4) for order of magnitude comparisons.

Our final assumption approximates the $\underline{\alpha}$ and $\underline{\omega}$ terms in (8.1.4.1.2-6) and the (8.1.4.1.2-2) $\underline{\alpha}$ expression, and the $\dot{\underline{\omega}}$ term in the (8.1.4.1.2-5) $\delta\underline{a}''_{Size}$ expression, by their equivalent $'$ values (without misalignment compensation). Then, $\delta\underline{v}_{Scul\text{-}SizeC_m}$ in (8.1.4.1.2-1) with (8.1.4.1.2-6) - (8.1.4.1.2-7), (8.1.4.1.2-2) for $\underline{\alpha}$, and (8.1.4.1.2-5) for $\delta\underline{a}''_{Size}$ and $\delta\underline{\upsilon}''_{SizeC}$ becomes;

$$\delta\underline{v}_{Scul\text{-}SizeC_m} \approx \int_{t_{m-1}}^{t_m} \frac{1}{2}\left[\underline{\alpha}' \times \delta\underline{a}''_{Size} + \delta\underline{\upsilon}''_{SizeC} \times \underline{\omega}'\right] dt \tag{8.1.4.1.2-8}$$

$$\delta\underline{a}''_{Size} \approx \sum_{k=1,3}\left[\underline{u}_k \cdot \left(\dot{\underline{\omega}}' \times \underline{l}_k\right)\right]\underline{u}_k$$
$$\underline{\alpha}' = \int_{t_{m-1}}^{t} \underline{\omega}'\, d\tau \qquad \delta\underline{\upsilon}''_{SizeC} = \int_{t_{m-1}}^{t} \delta\underline{a}''_{Size}\, d\tau \tag{8.1.4.1.2-9}$$

The direct algorithmic implementation of (8.1.4.1.2-8) and (8.1.4.1.2-9) implies a high speed integration within the m cycle (i.e., at the *l* cycle rate). The terms to be integrated are in the form of $\underline{\alpha}'$ products with $\dot{\underline{\omega}}'$ in the $\underline{\alpha}' \times \delta\underline{a}''_{Size}$ expression, and $\underline{\omega}'$ products with $\underline{\omega}'$ in $\delta\underline{\upsilon}''_{SizeC} \times \underline{\omega}'$. The $\underline{\omega}'$ with $\underline{\omega}'$ products are already calculated as part of the $\delta\underline{\upsilon}'_{SizeC_m}$ algorithm (in the form of Equations (8.1.4.1.1.1-14)). To eliminate an additional high speed calculation for $\underline{\alpha}'$ with $\dot{\underline{\omega}}'$ products, Equation (8.1.4.1.2-8) is modified by incorporating the following for the $\underline{\alpha}' \times \delta\underline{a}''_{Size}$ integrand term based on (8.1.4.1.2-9) for $\underline{\alpha}'$ and $\delta\underline{\upsilon}''_{SizeC}$, and the derivative of $\left(\underline{\alpha} \times \delta\underline{\upsilon}_{SizeC}\right)$, rearranged:

$$\underline{\alpha}' \times \delta\underline{a}''_{Size} = \frac{d}{dt}\left(\underline{\alpha}' \times \delta\underline{\upsilon}''_{SizeC}\right) - \underline{\omega}' \times \delta\underline{\upsilon}''_{SizeC} \tag{8.1.4.1.2-10}$$

Substituting (8.1.4.1.2-10) in (8.1.4.1.2-8) obtains the equivalent form:

$$\begin{aligned}\delta\underline{v}_{Scul\text{-}SizeC_m} &= \frac{1}{2}\left(\underline{\alpha}'_m \times \delta\underline{\upsilon}''_{SizeC_m}\right) - \int_{t_{m-1}}^{t_m}\left(\underline{\omega}' \times \delta\underline{\upsilon}''_{SizeC}\right) dt \\ &\approx \frac{1}{2}\left(\underline{\alpha}_m \times \delta\underline{\upsilon}''_{SizeC_m}\right) - \int_{t_{m-1}}^{t_m}\left(\underline{\omega}' \times \delta\underline{\upsilon}''_{SizeC}\right) dt\end{aligned} \tag{8.1.4.1.2-11}$$

Equation (8.1.4.1.2-11) is the form we will now use to find a digital computation algorithm for $\delta\underline{v}_{Scul\text{-}SizeC_m}$. Using (8.1.4.1.1.1-1) and (8.1.4.1.1.1-3) for component definitions, Equation (8.1.4.1.2-11) with (8.1.4.1.2-9) obtains for the $\delta\underline{v}_{Scul\text{-}SizeC_m}$ Y axis component (as representative):

$$\begin{aligned}\delta v_{Scul\text{-}SizeCY_m} = {} & \frac{1}{2}\,\alpha_{Z_m}\Big[\big(\omega'_{Y_m} + \omega'_{Y_{m-1}}\big)\, l_{Z_1} - \big(\omega'_{Z_m} + \omega'_{Z_{m-1}}\big)\, l_{Y_1}\Big] \\ & - \frac{1}{2}\,\alpha_{X_m}\Big[\big(\omega'_{X_m} + \omega'_{X_{m-1}}\big)\, l_{Y_3} - \big(\omega'_{Y_m} + \omega'_{Y_{m-1}}\big)\, l_{X_3}\Big] \qquad (8.1.4.1.2\text{-}12) \\ & + \gamma_{XX_m}\, l_{Y_3} + \gamma_{ZZ_m}\, l_{Y_1} - \gamma_{XY_m}\, l_{X_3} - \gamma_{YZ_m}\, l_{Z_1}\end{aligned}$$

Using Equations (8.1.4.1.1.1-10) and (8.1.4.1.1.1-12) for γ_{ij_m} and ω'_{i_m} in (8.1.4.1.2-12), then yields the $\delta\underline{v}_{Scul\text{-}SizeC_m}$ digital computation algorithm:

$\eta_{ij_m}, \Delta\alpha'_{j_m}$ = From Algorithm Equations (8.1.4.1.1.1-14).

α_{i_m} = From Algorithm Equation (8.1.2.1-4).

$$\begin{aligned}\delta v_{Scul\text{-}SizeCY_m} = {} & f_{Size}\Big\{\frac{1}{2}\,\alpha_{Z_m}\Big[\big(\Delta\alpha'_{Y_m} + \Delta\alpha'_{Y_{m-1}}\big)\, l_{Z_1} - \big(\Delta\alpha'_{Z_m} + \Delta\alpha'_{Z_{m-1}}\big)\, l_{Y_1}\Big] \\ & - \frac{1}{2}\,\alpha_{X_m}\Big[\big(\Delta\alpha'_{X_m} + \Delta\alpha'_{X_{m-1}}\big)\, l_{Y_3} - \big(\Delta\alpha'_{Y_m} + \Delta\alpha'_{Y_{m-1}}\big)\, l_{X_3}\Big] \\ & + \eta_{XX_m}\, l_{Y_3} + \eta_{ZZ_m}\, l_{Y_1} - \eta_{XY_m}\, l_{X_3} - \eta_{YZ_m}\, l_{Z_1}\Big\} \qquad (8.1.4.1.2\text{-}13)\end{aligned}$$

$\delta v_{Scul\text{-}SizeCZ_m}, \delta v_{Scul\text{-}SizeCX_m}$ = Similarly by permuting subscripts.

where

$\delta v_{Scul\text{-}SizeCX_m}, \delta v_{Scul\text{-}SizeCY_m}, \delta v_{Scul\text{-}SizeCZ_m}$ = B Frame X, Y, Z components of $\delta\underline{v}_{Scul\text{-}SizeC_m}$.

8.1.4.1.3 $\delta\underline{v}_{Rot/Scul\text{-}SizeC_m}$ Size Effect Algorithm

Given that the $\delta\underline{v}_{Scul\text{-}SizeC_m}$ algorithm has been derived in Section 8.1.4.1.2, it is tempting to short circuit the development process for the $\delta\underline{v}_{Rot/Scul\text{-}SizeC_m}$ algorithm by jumping directly to an approximation to (8.1.4.1-20) modeled after (8.1.4.1.2-8) for $\delta\underline{v}_{Scul\text{-}SizeC_m}$ (that was based on (8.1.4.1-15)). The quick result would approximate $\delta\underline{v}_{Rot/Scul\text{-}SizeC_m}$ as

$\int_{t_{m-1}}^{t_m} \left(\underline{\alpha}' \times \delta\underline{a}''_{Size}\right) dt$. Unfortunately, this approximation is flawed because it depends on mutual cancellation of low frequency $\underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k\right)$ terms contained in $\delta\underline{a}_{Size}$ of Equation (8.1.4.1-9) (as in Equation (8.1.4.1.2-7) for the $\delta\underline{v}_{Scul\text{-}SizeC_m}$ derivation). In this case, there is no $\delta\underline{\upsilon}_{SizeC_{Lo\text{-}f/\omega^2}} \times \underline{\omega}_{Lo\text{-}f}$ term to cancel $\underline{\alpha}_{Lo\text{-}f} \times \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}}$ in $\underline{\alpha} \times \delta\underline{a}_{Size}$ of $\delta\underline{v}_{Rot/Scul\text{-}SizeC_m}$ Equation (8.1.4.1-20). Thus, the previous approximation contains $\underline{\alpha}_{Lo\text{-}f} \times \delta\underline{a}_{Size_{Lo\text{-}f/\omega^2}}$ errors under high angular rate conditions that are not negligible for the Section 8.1.4.1.2 order of magnitude comparisons. If we are to include $\underline{\omega} \times \left(\underline{\omega} \times \underline{l}_k\right)$ in $\delta\underline{a}_{Size}$, a brute force implementation of (8.1.4.1-20) for $\delta\underline{v}_{Rot/Scul\text{-}SizeC_m}$ would contain high speed (l cycle) integrations of terms of the form $\alpha_i\ \omega_j\ \omega_k$, terms not yet contained in any of the size effect algorithms developed thus far. To avoid the additional high speed integrations, the alternative derivation approach described below is taken for the $\delta\underline{v}_{Rot/Scul\text{-}SizeC_m}$ algorithm.

The $\delta\underline{v}_{Rot/Scul\text{-}SizeC_m}$ algorithm derivation begins with approximating the G_{Algn} matrix in (8.1.4.1-20) by identity so that (8.1.4.1-20) becomes:

$$\delta\underline{v}_{Rot/Scul\text{-}SizeC_m} \approx \int_{t_{m-1}}^{t_m} \left(\underline{\alpha} \times \delta\underline{a}_{Size}\right) dt \tag{8.1.4.1.3-1}$$

Using the same approximation, we also know from (8.1.4.1.2-2), (8.1.4.1-11) and the derivative of $\left(\underline{\alpha} \times \delta\underline{\upsilon}_{SizeC}\right)$ that:

$$\underline{\alpha} = \int_{t_{m-1}}^{t} \underline{\omega}\, d\tau \qquad \delta\underline{\upsilon}_{SizeC} \approx \int_{t_{m-1}}^{t} \delta\underline{a}_{Size} d\tau \tag{8.1.4.1.3-2}$$

$$\begin{aligned} \frac{1}{2}\,\underline{\alpha} \times \delta\underline{a}_{Size} &= \frac{1}{2}\,\frac{d}{dt}\left(\underline{\alpha} \times \delta\underline{\upsilon}_{SizeC}\right) - \frac{1}{2}\,\underline{\omega} \times \delta\underline{\upsilon}_{SizeC} \\ &= \frac{1}{2}\,\frac{d}{dt}\left(\underline{\alpha} \times \delta\underline{\upsilon}_{SizeC}\right) + \frac{1}{2}\,\delta\underline{\upsilon}_{SizeC} \times \underline{\omega} \end{aligned} \tag{8.1.4.1.3-3}$$

hence,

$$\begin{aligned} \underline{\alpha} \times \delta\underline{a}_{Size} &= \frac{1}{2}\,\underline{\alpha} \times \delta\underline{a}_{Size} + \frac{1}{2}\,\underline{\alpha} \times \delta\underline{a}_{Size} \\ &= \frac{1}{2}\,\frac{d}{dt}\left(\underline{\alpha} \times \delta\underline{\upsilon}_{SizeC}\right) + \frac{1}{2}\left(\underline{\alpha} \times \delta\underline{a}_{Size} + \delta\underline{\upsilon}_{SizeC} \times \underline{\omega}\right) \end{aligned} \tag{8.1.4.1.3-4}$$

Substituting (8.1.4.1.3-4) in (8.1.4.1.3-1) then finds:

$$\delta \underline{v}_{Rot/Scul\text{-}SizeC_m} = \frac{1}{2}\left(\underline{\alpha}_m \times \delta \underline{\upsilon}_{SizeC_m}\right) + \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha} \times \delta \underline{a}_{Size} + \delta \underline{\upsilon}_{SizeC} \times \underline{\omega}\right) dt \qquad (8.1.4.1.3\text{-}5)$$

or with the integral term recognized as $\delta \underline{v}_{Scul\text{-}SizeC_m}$ from (8.1.4.1.2-1):

$$\delta \underline{v}_{Rot/Scul\text{-}SizeC_m} = \frac{1}{2}\left(\underline{\alpha}_m \times \delta \underline{\upsilon}_{SizeC_m}\right) + \delta \underline{v}_{Scul\text{-}SizeC_m} \qquad (8.1.4.1.3\text{-}6)$$

The digital algorithm for $\delta \underline{v}_{Rot/Scul\text{-}SizeC_m}$ is then found by substituting for $\delta \underline{v}_{Scul\text{-}SizeC_m}$ from (8.1.4.1.2-13):

$\eta_{ij_m}, \Delta\alpha'_{j_m}$ = From Algorithm Equations (8.1.4.1.1.1-14).

α_{i_m} = From Algorithm Equation (8.1.2.1-4).

$\delta\upsilon_{SizeCi_m}$ = From Algorithm Equations (8.1.4.1.1-11) With (8.1.4.1.1.1-15) And (8.1.4.1.1.2-3).

$$\begin{aligned} \delta v_{Rot/Scul\text{-}SizeCY_m} = & \frac{1}{2}\left(\alpha_{Z_m}\, \delta\upsilon_{SizeCX_m} - \alpha_{X_m}\, \delta\upsilon_{SizeCZ_m}\right) \qquad (8.1.4.1.3\text{-}7) \\ & + f_{Size}\left\{ \frac{1}{2}\, \alpha_{Z_m}\left[\left(\Delta\alpha'_{Y_m} + \Delta\alpha'_{Y_{m-1}}\right) l_{Z_1} - \left(\Delta\alpha'_{Z_m} + \Delta\alpha'_{Z_{m-1}}\right) l_{Y_1}\right] \right. \\ & \quad - \frac{1}{2}\, \alpha_{X_m}\left[\left(\Delta\alpha'_{X_m} + \Delta\alpha'_{X_{m-1}}\right) l_{Y_3} - \left(\Delta\alpha'_{Y_m} + \Delta\alpha'_{Y_{m-1}}\right) l_{X_3}\right] \\ & \quad \left. + \eta_{XX_m}\, l_{Y_3} + \eta_{ZZ_m}\, l_{Y_1} - \eta_{XY_m}\, l_{X_3} - \eta_{YZ_m}\, l_{Z_1} \right\} \end{aligned}$$

$\delta v_{Rot/Scul\text{-}SizeCX_m}, \delta v_{Rot/Scul\text{-}SizeCZ_m}$ = Similarly by permuting subscripts.

where

$\delta\upsilon_{SizeCi_m}$ = i^{th} component of $\delta\underline{\upsilon}_{SizeC_m}$.

$\delta v_{Rot/Scul\text{-}SizeCX_m}, \delta v_{Rot/Scul\text{-}SizeCY_m}, \delta v_{Rot/Scul\text{-}SizeCZ_m}$ = B Frame X, Y, Z components of $\delta \underline{v}_{Rot/Scul\text{-}SizeC_m}$.

For most applications, $\delta\upsilon_{SizeCi_m}$ in (8.1.4.1.3-7) can be approximated by the equivalent $\delta\upsilon'_{SizeCi_m}$ form in (8.1.4.1.1.1-15) that neglects sensor misalignment effects. This is equivalent to neglecting the $\Delta\delta\underline{\upsilon}_{SizeC_m}$ term in $\delta\underline{\upsilon}_{SizeC_m}$ Equation (8.1.4.1.1-11).

8.1.4.1.4 Sensor Compensation Applied To Size Effect Algorithm Terms

The l loop summation and sampling operations in size effect Equations (8.1.4.1.1.1-14) can be performed using raw (unscaled and uncompensated) angular rate sensor outputs with scaling and compensation then applied at the acceleration-transformation/velocity-update time t_m. The raw output summation and sampling operations would then have the form:

$$\Delta\alpha_{iCnt_l} \equiv \int_{t_{l-1}}^{t_l} d\alpha_{iCnt} \qquad \text{Summation Increment Of Angular Rate Sensor Output Pulses}$$

$$\eta_{ijCnt_l} = \eta_{ijCnt_{l-1}} + \Delta\alpha_{iCnt_l}\,\Delta\alpha_{jCnt_l}$$

$$\eta_{ijCnt_m} = \eta_{ijCnt_l}(t_l = t_m) \qquad \eta_{ijCnt_l} = 0 \quad \text{At} \;\; t = t_{m-1} \tag{8.1.4.1.4-1}$$

$$\Delta\alpha_{iCnt_m} = \Delta\alpha_{iCnt_l} \quad \text{At} \;\; t = t_m$$

$$\Delta\alpha_{iCnt_{(l\,:\,m)-1}} = \Delta\alpha_{iCnt_l} \quad \text{Immediately preceding} \;\; t = t_m.$$

where

Cnt = Label for Equations (8.1.4.1.1.1-14) parameters computed with unscaled and uncompensated angular rate sensor outputs.

$d\alpha_{iCnt} = \omega_{iPuls}\, dt$ = Unscaled differential integrated i axis angular pulse rate increment (i.e., analytical representation of raw pulse output from the i axis strapdown angular rate sensor).

The Equation (8.1.4.1.4-1) results can be scaled using the first Equation (8.1.2.1-2) compensation formula:

$$\underline{\omega}' = \Omega_{Wt}\, \underline{\omega}_{Puls} \tag{8.1.4.1.4-2}$$

Applying (8.1.4.1.4-2) to Equations (8.1.4.1.4-1) using (8.1.4.1.1.1-9) for η_{ij_m} definition yields the scaled version of the size effect algorithm parameters:

$$\eta_{ij_m} = \Omega_{Wt_i}\,\Omega_{Wt_j}\,\eta_{ijCnt_m}$$

$$\Delta\alpha'_{i_m} = \Omega_{Wt_i}\,\Delta\alpha_{iCnt_m} \tag{8.1.4.1.4-3}$$

$$\Delta\alpha'_{i_{(l\,:\,m)-1}} = \Omega_{Wt_i}\,\Delta\alpha_{iCnt_{(l\,:\,m)-1}}$$

where

$'$ = Reference to scaled parameters, but compensated only for scale factor error. Note that η_{ij_m} is already defined by (8.1.4.1.1.1-9) to be based on scale factor only compensation.

Ω_{Wt_i} = Element in row i, column i of diagonal matrix Ω_{Wt}.

Equations (8.1.4.1.4-1) and (8.1.4.1.4-3) for η_{ij_m}, $\Delta\alpha'_{i_m}$, $\Delta\alpha'_{i_{(l\,:\,m)-1}}$ are based on the assumption that Ω_{Wt_i} can be approximated as constant over an m cycle. Under this assumption, if Ω_{Wt_i} contains non-linear terms (as addressed in Section 8.1.1.3), Ω_{Wt_i} would be calculated as a function of α_{iCnt_m}, the sum of the $\Delta\alpha_{iCnt_l}$'s over an m cycle (See Equation (8.2.1.1-15) for $\underline{\alpha}_{Cnt_m}$ algorithm). For the more general case in which Ω_{Wt_i} may have rapid variations, η_{ij_m}, $\Delta\alpha'_{i_m}$, $\Delta\alpha'_{i_{(l\,:\,m)-1}}$ can be calculated based on Equation (8.1.4.1.1.1-3), but computed from scale factor compensated angular rate sensor output data $\underline{\omega}'$ as in (8.1.4.1.4-2). Then the equivalent to (8.1.4.1.4-1) and (8.1.4.1.4-3) for η_{ij_m}, $\Delta\alpha'_{i_m}$, $\Delta\alpha'_{i_{(l\,:\,m)-1}}$ would be:

$$\Delta\alpha_{iCnt_l} \equiv \int_{t_{l-1}}^{t_l} d\alpha_{iCnt} \qquad \text{Summation Increment Of Angular Rate Sensor Output Pulses}$$

$$\Delta\alpha'_{i_l} = \Omega_{Wt_i}\,\Delta\alpha_{iCnt_l} \tag{8.1.4.1.4-4}$$

$$\eta_{ij_l} = \eta_{ij_{l-1}} + \Delta\alpha'_{i_l}\,\Delta\alpha'_{j_l}$$

$$\eta_{ij_m} = \eta_{ij_l}(t_l = t_m) \qquad \eta_{ij_l} = 0 \quad \text{At } t = t_{m-1} \tag{8.1.4.1.4-5}$$

$$\Delta\alpha'_{i_m} = \Delta\alpha'_{i_l} \quad \text{At } t = t_m$$

$$\Delta\alpha'_{i_{(l\,:\,m)-1}} = \Delta\alpha'_{i_l} \quad \text{Immediately preceding } t = t_m.$$

If scale factor asymmetry compensation is being applied in the software algorithms as described in Section 8.1.1.3, Equation (8.1.4.1.4-4) would be based on $\underline{\omega}'$ given by Equation (8.1.1.3-21) rather than (8.1.4.1.4-2):

$$\underline{\omega}' = \Omega_{Wt_+}\,\underline{\omega}_{+Puls} + \Omega_{Wt_-}\,\underline{\omega}_{-Puls} \tag{8.1.4.1.4-6}$$

The Ω_{Wt_+}, Ω_{Wt_-} terms in Equation (8.1.4.1.4-6) are scale factor weighting matrices defined in Equation (8.1.1.3-22) for positive and negative uncompensated angular rate sensor pulse rate output data ($\underline{\omega}_{+Puls}$ and $\underline{\omega}_{-Puls}$ - See Section 8.1.1.3 for further clarification). Based on Equation (8.1.4.1.4-6), the equivalent to (8.1.4.1.4-4) would be as follows:

$$\Delta\alpha_{i+Cnt_l} \equiv \int_{t_{l-1}}^{t_l} d\alpha_{i+Cnt} \qquad \text{Summation Increment Of Positive Angular Rate Sensor Output Pulses}$$

$$\Delta\alpha_{i-Cnt_l} \equiv \int_{t_{l-1}}^{t_l} d\alpha_{i-Cnt} \qquad \text{Summation Increment Of Negative Angular Rate Sensor Output Pulses} \tag{8.1.4.1.4-7}$$

$$\Delta\alpha'_{i_l} = \Omega_{Wt\ i+}\,\Delta\alpha_{i+Cnt_l} + \Omega_{Wt\ i-}\,\Delta\alpha_{i-Cnt_l}$$

where

$d\alpha_{i+Cnt}, d\alpha_{i-Cnt}$ = $\omega_{i+Puls}\,dt$ and $\omega_{i-Puls}\,dt$, the uncompensated i axis angular rate sensor differential positive and negative pulse rate outputs.

$\Omega_{Wt\ i+}, \Omega_{Wt\ i-}$ = Angular rate sensor i positive and negative pulse scale factors.

Equations (8.1.4.1.4-7) are based on the assumption that separate plus and minus pulse sum outputs are provided from the angular rate sensor interface to the navigation software. If only a composite pulse sum is available (as in (8.1.4.1.4-4)), Equation (8.1.4.1.4-7) would be of the form:

$$\Delta\alpha_{iCnt_l} \equiv \int_{t_{l-1}}^{t_l} d\alpha_{iCnt} \qquad \text{Summation Increment Of Angular Rate Sensor Output Pulses}$$

$$\text{If } \Delta\alpha_{iCnt_l} \geq 0 \quad \text{Then:} \quad \Delta\alpha'_{i_l} = \Omega_{Wt\ i+}\,\Delta\alpha_{iCnt_l} \tag{8.1.4.1.4-8}$$

$$\text{Else:} \quad \Delta\alpha'_{i_l} = \Omega_{Wt\ i-}\,\Delta\alpha_{iCnt_l}$$

The η_{ij_m}, $\Delta\alpha'_{i_m}$, $\Delta\alpha'_{i_{(l\,:\,m)\,-1}}$ data from (8.1.4.1.4-1) and (8.1.4.1.4-3), or from (8.1.4.1.4-4) (or (8.1.4.1.4-7)) and (8.1.4.1.4-5), can be used directly in the Equation (8.1.4.1.1.1-15), (8.1.4.1.1.2-3), (8.1.4.1.2-13) and (8.1.4.1.3-7) size effect algorithms, and in quantization algorithm (8.1.3.3-5).

For higher accuracy, quantization compensation can be added in the above $\Delta\alpha'_{i_m}$ calculations based on an approximation to the second expression in Equations (8.1.2.1-2):

$$\underline{\omega}'^{Q} = \underline{\omega}' - \delta\underline{\omega}_{Quant} \tag{8.1.4.1.4-9}$$

where

$\underline{\omega}'^{Q}$ = Enhanced accuracy version of $\underline{\omega}'$ that includes both scale factor and quantization compensation.

Applying (8.1.4.1.4-9) to $\Delta\underline{\alpha}'_{i_m}$ in (8.1.4.1.1.1-14) yields the enhanced accuracy version:

$$\Delta\alpha'^{Q}_{i_m} = \Delta\alpha'_{i_m} - \delta\underline{\alpha}_{QuantC_{l:m}} \qquad (8.1.4.1.4\text{-}10)$$

where

$\Delta\alpha'^{Q}_{i_m}$, = Enhanced accuracy version of $\Delta\underline{\alpha}'_{i_m}$ that includes both scale factor and quantization compensation.

$\delta\underline{\alpha}_{QuantC_{l:m}}$ = Integrated angular rate sensor quantization error over the l cycle time period, evaluated at the acceleration-transformation/velocity-update time t_m.

Section 8.1.3.3 provides formulas for calculating $\delta\underline{\alpha}_{QuantC_{l:m}}$ in Equation (8.1.4.1.4-10).

Extension of the same technique for the adjunct to (8.1.4.1.4-1) and (8.1.4.1.4-5) should be obvious.

8.1.4.1.5 Size Effect Algorithms Under Benign Environments

Under benign environments (i.e., slowly changing angular rates) the high speed size effect l cycle calculations can be executed at the m cycle acceleration-transformation/velocity-update rate. Under these conditions $\Delta\alpha'_{i_m} = \alpha'_{i_m} \approx \alpha_{i_m}$ and Equations (8.1.4.1.1.1-15), (8.1.4.1.1.2-3), (8.1.4.1.1.2-5), (8.1.4.1.2-13) and (8.1.4.1.3-7) become:

$$\begin{aligned}\delta\upsilon'_{SizeCY_m} \approx\ & f_{Size}\Big[-\, l_{Z_2}\left(\alpha_{X_m} - \alpha_{X_{m-1}}\right) + l_{X_2}\left(\alpha_{Z_m} - \alpha_{Z_{m-1}}\right) \\ & + l_{Z_2}\,\alpha_{Y_m}\,\alpha_{Z_m} + l_{X_2}\,\alpha_{X_m}\,\alpha_{Y_m} - l_{Y_2}\left(\alpha_{Z_m}\,\alpha_{Z_m} + \alpha_{X_m}\,\alpha_{X_m}\right)\Big]\end{aligned} \qquad (8.1.4.1.5\text{-}1)$$

$\delta\upsilon'_{SizeCZ_m}, \delta\upsilon'_{SizeCX_m}$ = Similarly by permuting subscripts.

$$\begin{aligned}\Delta\delta\upsilon_{SizeCY_m} \approx\ & -\, L_{Mis_{YX}}\left(\delta\upsilon'_{SizeCX_m} - \delta\upsilon'_{SizeCX/2_m}\right) - L_{Mis_{YZ}}\left(\delta\upsilon'_{SizeCZ_m} - \delta\upsilon'_{SizeCZ/2_m}\right) \\ & + f_{Size}\Big\{-\, K_{Mis_{ZX}}\, l_{X_2}\left(\alpha_{X_m} - \alpha_{X_{m-1}}\right) - \left(K_{Mis_{ZY}}\, l_{X_2} - K_{Mis_{XY}}\, l_{Z_2}\right)\left(\alpha_{Y_m} - \alpha_{Y_{m-1}}\right) \\ & + K_{Mis_{XZ}}\, l_{Z_2}\left(\alpha_{Z_m} - \alpha_{Z_{m-1}}\right) - \left(K_{Mis_{YX}}\, l_{X_2}\right)\alpha^2_{X_m} \\ & - \left(K_{Mis_{XY}}\, l_{X_2} + K_{Mis_{ZY}}\, l_{Z_2}\right)\alpha^2_{Y_m} - \left(K_{Mis_{YZ}}\, l_{Z_2}\right)\alpha^2_{Z_m}\end{aligned} \qquad (8.1.4.1.5\text{-}2)$$

(Continued)

$$+ \left(2\, K_{Mis_{XY}}\, l_{Y_2} - K_{Mis_{ZX}}\, l_{Z_2}\right) \alpha_{X_m}\, \alpha_{Y_m} + \left(2\, K_{Mis_{ZY}}\, l_{Y_2} - K_{Mis_{XZ}}\, l_{X_2}\right) \alpha_{Y_m}\, \alpha_{Z_m}$$
$$+ \left[2 \left(K_{Mis_{XZ}} + K_{Mis_{ZX}}\right) l_{Y_2} - K_{Mis_{YZ}}\, l_{X_2} - K_{Mis_{YX}}\, l_{Z_2}\right] \alpha_{Z_m}\, \alpha_{X_m} \Big\}$$

(8.1.4.1.5-2) (Continued)

$\Delta\delta\upsilon_{SizeCZ_m}, \Delta\delta\upsilon_{SizeCX_m}$ = Similarly by permuting subscripts.

$$\delta\upsilon'_{SizeCX/k_m} = f_{Size} \Big[- l_{Y_k} \left(\alpha_{Z_m} - \alpha_{Z_{m-1}}\right) + l_{Z_k} \left(\alpha_{Y_m} - \alpha_{Y_{m-1}}\right)$$
$$+ l_{Y_k}\, \alpha_{X_m}\, \alpha_{Y_m} + l_{Z_k}\, \alpha_{Z_m}\, \alpha_{X_m} - l_{X_k} \left(\alpha_{Y_m}^2 + \alpha_{Z_m}^2\right) \Big] \quad (8.1.4.1.5\text{-}3)$$

$\delta\upsilon'_{SizeCY/k_m}, \delta\upsilon'_{SizeCZ/k_m}$ = Similarly by permuting subscripts.

$$\delta v_{Scul\text{-}SizeCY_m} \approx f_{Size} \Big\{ \frac{1}{2} \alpha_{Z_m} \Big[\left(\alpha_{Y_m} + \alpha_{Y_{m-1}}\right) l_{Z_1} - \left(\alpha_{Z_m} + \alpha_{Z_{m-1}}\right) l_{Y_1} \Big]$$
$$- \frac{1}{2} \alpha_{X_m} \Big[\left(\alpha_{X_m} + \alpha_{X_{m-1}}\right) l_{Y_3} - \left(\alpha_{Y_m} + \alpha_{Y_{m-1}}\right) l_{X_3} \Big] \quad (8.1.4.1.5\text{-}4)$$
$$+ \alpha_{X_m}^2\, l_{Y_3} + \alpha_{Z_m}^2\, l_{Y_1} - \alpha_{X_m}\, \alpha_{Y_m}\, l_{X_3} - \alpha_{Y_m}\, \alpha_{Z_m}\, l_{Z_1} \Big\}$$

$\delta v_{Scul\text{-}SizeCZ_m}, \delta v_{Scul\text{-}SizeCX_m}$ = Similarly by permuting subscripts.

$$\delta v_{Rot/Scul\text{-}SizeCY_m} \approx \frac{1}{2} \left(\alpha_{Z_m}\, \delta\upsilon_{SizeCX_m} - \alpha_{X_m}\, \delta\upsilon_{SizeCZ_m}\right)$$
$$+ f_{Size} \Big\{ \frac{1}{2} \alpha_{Z_m} \Big[\left(\alpha_{Y_m} + \alpha_{Y_{m-1}}\right) l_{Z_1} - \left(\alpha_{Z_m} + \alpha_{Z_{m-1}}\right) l_{Y_1} \Big]$$
$$- \frac{1}{2} \alpha_{X_m} \Big[\left(\alpha_{X_m} + \alpha_{X_{m-1}}\right) l_{Y_3} - \left(\alpha_{Y_m} + \alpha_{Y_{m-1}}\right) l_{X_3} \Big] \quad (8.1.4.1.5\text{-}5)$$
$$+ \alpha_{X_m}^2\, l_{Y_3} + \alpha_{Z_m}^2\, l_{Y_1} - \alpha_{X_m}\, \alpha_{Y_m}\, l_{X_3} - \alpha_{Y_m}\, \alpha_{Z_m}\, l_{Z_1} \Big\}$$

$\delta v_{Rot/Scul\text{-}SizeCX_m}, \delta v_{Rot/Scul\text{-}SizeCZ_m}$ = Similarly by permuting subscripts.

The α terms in Equations (8.1.4.1.5-1) - (8.1.4.1.5-5) would be the compensated values provided by Equations (8.1.2.1-3) (or (8.1.2.1-7)) and (8.1.2.1-4).

8.1.4.2 PENDULOUS ACCELEROMETER ANISOINERTIA COMPENSATION ALGORITHM

Pendulous accelerometers have a dynamic error characteristic denoted as anisoinertia error which is proportional to the product of the angular rates along the input and pendulum axes (See Reference 31):

$$\delta a_{Aniso_k} = K_{Aniso}\, \omega_k\, \omega_p \tag{8.1.4.2-1}$$

where

δa_{Aniso_k} = Accelerometer k anisoinertia error.

ω_k, ω_p = Angular rates around accelerometer k input axis (k) and pendulum axis (p).

K_{Aniso} = Accelerometer anisoinertia coefficient.

The integral of Equation (8.1.4.2-1) (multiplied by the kth row of G_{Algn}^{-1} for the k axis accelerometer - as in (8.1.2.2-2)) is used to compensate the integrated specific force acceleration output from the accelerometer, as indicated by $\delta\underline{\upsilon}_{AnisoC_m}$ in Equations (8.1.2.2-4) and (8.1.2.2-6). If we make the approximation of neglecting inertial sensor misalignment effects, the components of $\delta\underline{\upsilon}_{AnisoC_m}$ become the simple integral of (8.1.4.2-1) using $\omega'_k\, \omega'_p$ for $\omega_k\, \omega_p$, or with (8.1.4.1.1.1-3) for the $\omega'_k\, \omega'_p$ integral:

$$\delta\upsilon_{AnisoCk_m} = K_{Aniso}\, \gamma_{kp_m} \tag{8.1.4.2-2}$$

where

$\delta\upsilon_{AnisoCk_m}$ = Component of $\delta\underline{\upsilon}_{AnisoC_m}$ for the k axis accelerometer.

or, using (8.1.4.1.1.1-10):

$$\delta\upsilon_{AnisoCk_m} = f_{Size}\, K_{Aniso}\, \eta_{kp_m} \tag{8.1.4.2-3}$$

with η_{kp_m} calculated from the (8.1.4.1.1.1-14) algorithm (or the (8.1.4.1.4-1) and (8.1.4.1.4-3) variants).

For enhanced accuracy, a correction term can be included in (8.1.4.2-3) that accounts for inertial sensor misalignments (as was the $\Delta\delta\underline{\upsilon}_{SizeC_m}$ sensor misalignment correction in (8.1.4.1.1-11) for the $\delta\underline{\upsilon}_{SizeC_m}$ algorithm). Since the K_{Aniso} coefficient is generally much smaller than the size effect lever arms, $\delta\underline{\upsilon}_{AnisoC_m}$ is generally smaller than $\delta\underline{\upsilon}_{SizeC_m}$, hence, a

$\delta \underline{\upsilon}_{AnisoC_m}$ misalignment correction is generally negligible compared to $\Delta\delta\underline{\upsilon}_{SizeC_m}$. It is also noted that the anisoinertia term affects the $\Delta \underline{v}_{Scul}$ sculling algorithm of Section 8.2.2.1 and the $\Delta \underline{v}_{Rot/Scul}$ combined rotation-compensation/sculling algorithm of Section 8.2.2.2, both sections to follow, which do not include anisoinertia compensation in their calculations. The associated error effect is generally negligible compared to the $\delta \underline{\upsilon}_{Scul\text{-}SizeC}$ or $\delta \underline{\upsilon}_{Rot/Scul\text{-}SizeC}$ correction terms of Sections 8.1.4.1.2 and 8.1.4.1.3 (corrections to $\Delta \underline{v}_{Scul}$ and $\Delta \underline{v}_{Rot/Scul}$), due to the generally smaller value of the K_{Aniso} coefficient compared to the size effect lever arms.

For benign angular rate environments, (8.1.4.2-2) can be approximated as in Equations (8.1.4.1.5-1) - (8.1.4.1.5-4):

$$\delta \upsilon_{AnisoCk_m} \approx K_{Aniso}\, \alpha_{k_m}\, \alpha_{p_m} \tag{8.1.4.2-4}$$

where

$\alpha_{k_m}, \alpha_{p_m}$ = Components of $\underline{\alpha}_m$ from Equations (8.1.2.1-3) - (8.1.2.1-4) along accelerometer k input (k) and pendulum (p) axes.

8.2 INERTIAL SENSOR COMPENSATION APPLIED TO NAVIGATION ALGORITHMS

Section 8.1 described algorithms for compensating the inertial sensor outputs for predictable error characteristics. In this section we apply the Section 8.1 sensor compensation to the strapdown inertial navigation attitude, velocity and position updating algorithms developed in Chapters 7 and 19 (Section 19.1). Included is a detailed discussion of methods for compensating the coning, sculling, and scrolling high speed algorithms for inertial sensor error.

8.2.1 INERTIAL SENSOR COMPENSATION FOR ATTITUDE UPDATING

The angular rate sensor outputs in a strapdown system are used through suitable algorithms (Section 7.1) to update the system attitude reference. In general, this is achieved by processing Equations (7.1.1.1-12) and (7.1.1.1-13) (repeated below) over an attitude update cycle to calculate a rotation vector. The rotation vector is then used to update the strapdown attitude data. The rotation vector so calculated is also identical to the rotation vector of Section 19.1, Equations (19.1.5-9) for updating velocity/position based on a new unified strapdown algorithm concept that uses velocity/position translation vectors (analogous to the rotation vector) obtained by integrating Equations (19.1.8-3).

$$\underline{\phi}_m = \underline{\alpha}_m + \underline{\beta}_m$$

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}\, d\tau \qquad\qquad \underline{\alpha}_m = \underline{\alpha}(t_m) \tag{8.2.1-1}$$

$$\underline{\beta}_m = \frac{1}{2}\int_{t_{m-1}}^{t_m} \left(\underline{\alpha}(t) \times \underline{\omega}\right) dt$$

where

$\underline{\omega}$ = Angular rate vector sensed by angular rate sensor triad.

m = Attitude update cycle index.

$\underline{\alpha}_m$ = Integrated angular rate sensor vector output from t_{m-1} to t_m.

$\underline{\beta}_m$ = Coning attitude motion from t_{m-1} to t_m .

$\underline{\phi}_m$ = Rotation vector over m^{th} attitude update interval.

All vectors in (8.2.1-1) are in B Frame coordinates, the superscript notation for which has been omitted for simplicity. Similarly, the IB subscript has been dropped from the $\underline{\omega}$ angular rate sensor vector which was identified in Chapter 7 as $\underline{\omega}_{IB}$.

Equations (8.2.1-1) are based on the use of compensated angular rate sensor input data. Equations (8.1.2.1-3) (or (8.1.2.1-7)) and (8.1.2.1-4) provide the means for calculating the compensated form of $\underline{\alpha}_m$ for Equations (8.2.1-1). The $\underline{\beta}_m$ coning term in (8.2.1-1) can be determined using compensated $\underline{\omega}$ angular rate input data based on compensation Equations (8.1.2.1-2). A problem with this approach is that the $\underline{\omega}$ compensation operations would have to be performed at the high rate used to calculate $\underline{\beta}_m$ (i.e., the l computer cycle rate as depicted in Equations (7.1.1.1.1-18) which is the computer implementation of (8.2.1-1)). Alternatively, the Equations (7.1.1.1.1-18) coning algorithm can be determined with uncompensated angular rate sensor data, with the uncompensated coning solution then compensated for angular rate sensor error at the attitude update rate (i.e., the m rate in Equations (7.1.1.1.1-18)). The following section describes the latter approach.

8.2.1.1 CONING INCREMENT COMPENSATION ALGORITHM

Development of the algorithm for compensating the Equation (8.2.1-1) $\underline{\beta}_m$ coning term begins with an approximate form of Equations (8.1.1.1-8) which neglects the quantization compensation terms as second order for coning increment compensation purposes:

$$\underline{\omega}' = \Omega_{Wt}\, \underline{\omega}_{Puls} \tag{8.2.1.1-1}$$

$$\underline{\omega} \approx \underline{\omega}' - K_{Mis}\, \underline{\omega}' - \underline{K}_{Bias} \tag{8.2.1.1-2}$$

where

$\underline{\omega}_{Puls}$ = Uncompensated angular rate sensor output pulse rate vector (pulses per second).

Ω_{Wt} = Diagonal angular rate sensor pulse weighting matrix (radians per pulse) defined in Equations (8.1.1.1-5).

K_{Mis}, $\underline{K}_{Bias}$ = Angular rate sensor misalignment matrix and bias vector compensation coefficients as defined in Equations (8.1.1.1-6) and (8.1.1.1-7).

Substituting (8.2.1.1-1) and (8.2.1.1-2) into the (8.2.1-1) $\underline{\alpha}(t)$ expression then obtains:

$$\underline{\alpha}'(t) = \Omega_{Wt}\, \underline{\alpha}_{Cnt}(t) \tag{8.2.1.1-3}$$

$$\underline{\alpha}_{Cnt}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{Puls}\, d\tau \tag{8.2.1.1-4}$$

$$\underline{\alpha}(t) \approx \underline{\alpha}'(t) - K_{Mis}\, \underline{\alpha}'(t) - \underline{K}_{Bias}\left(t - t_{m-1}\right) \tag{8.2.1.1-5}$$

where

$\underline{\alpha}_{Cnt}(t)$ = $\underline{\alpha}(t)$ computed from uncompensated angular rate sensor triad output data.

We now substitute (8.2.1.1-2) and (8.2.1.1-5) into the (8.2.1-1) $\underline{\beta}_m$ expression to obtain for the integrand:

$$\begin{aligned}
\dot{\underline{\beta}}(t) &= \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{\omega}\right) \\
&= \frac{1}{2}\left[\underline{\alpha}'(t) - K_{Mis}\, \underline{\alpha}'(t) - \underline{K}_{Bias}\left(t - t_{m-1}\right)\right] \times \left[\underline{\omega}' - K_{Mis}\, \underline{\omega}' - \underline{K}_{Bias}\right] \\
&\approx \frac{1}{2}\Big[\underline{\alpha}'(t) \times \underline{\omega}' - \left(K_{Mis}\, \underline{\alpha}'(t)\right) \times \underline{\omega}' - \underline{\alpha}'(t) \times \left(K_{Mis}\, \underline{\omega}'\right) \\
&\qquad - \left(\underline{\alpha}'(t) - \underline{\omega}'\left(t - t_{m-1}\right)\right) \times \underline{K}_{Bias}\Big] \\
&\approx \frac{1}{2}\left[\underline{\alpha}'(t) \times \underline{\omega}' - \left(K_{Mis}\, \underline{\alpha}'(t)\right) \times \underline{\omega}' - \underline{\alpha}'(t) \times \left(K_{Mis}\, \underline{\omega}'\right)\right]
\end{aligned} \tag{8.2.1.1-6}$$

Equation (8.2.1.1-6) is based on dropping K_{Mis}, $\underline{K}_{Bias}$ product terms as second order and approximating $\underline{\omega}'$ as constant in the $\left(\underline{\alpha}'(t) - \underline{\omega}'\left(t - t_{m-1}\right)\right) \times \underline{K}_{Bias}$ term so that $\underline{\alpha}'(t) \approx \underline{\omega}'\left(t - t_{m-1}\right)$.

By component expansion and grouping of terms it easily demonstrated that Equation (8.2.1.1-6) can be written in the equivalent form:

$$\dot{\underline{\beta}}(t) = \left(I - K_{MisCone}\right) \dot{\underline{\beta}}'(t) \tag{8.2.1.1-7}$$

with

$$\dot{\underline{\beta}}'(t) = \frac{1}{2}\left(\underline{\alpha}'(t) \times \underline{\omega}'\right) \tag{8.2.1.1-8}$$

$$K_{MisCone} = \begin{bmatrix} \left(K_{Mis_{YY}} + K_{Mis_{ZZ}}\right) & - K_{Mis_{YX}} & - K_{Mis_{ZX}} \\ - K_{Mis_{XY}} & \left(K_{Mis_{ZZ}} + K_{Mis_{XX}}\right) & - K_{Mis_{ZY}} \\ - K_{Mis_{XZ}} & - K_{Mis_{YZ}} & \left(K_{Mis_{XX}} + K_{Mis_{YY}}\right) \end{bmatrix} \tag{8.2.1.1-9}$$

where

$\dot{\underline{\beta}}'(t) = \dot{\underline{\beta}}(t)$ computed with scaled but uncompensated inertial sensor data (except for scale factor error correction which is included in the scaling operation).

$K_{Mis_{ij}}$ = Element of K_{Mis} in row i and column j.

From the definition of K_{Mis} in Equation (8.1.1.1-6), it can be demonstrated that the diagonal elements are second order compared to the off-diagonal terms. This forms the basis for neglecting the K_{Mis} diagonal elements in Equation (8.2.1.1-9) to obtain the simpler form:

$$K_{MisCone} \approx \begin{bmatrix} 0 & - K_{Mis_{YX}} & - K_{Mis_{ZX}} \\ - K_{Mis_{XY}} & 0 & - K_{Mis_{ZY}} \\ - K_{Mis_{XZ}} & - K_{Mis_{YZ}} & 0 \end{bmatrix} \tag{8.2.1.1-10}$$

Equation (8.2.1.1-8) can be defined in terms of the equivalent uncompensated coning rate term by substitution of (8.2.1.1-1) and (8.2.1.1-3) into (8.2.1.1-8), including (8.2.1.1-4), expansion and regrouping of terms:

$$\dot{\underline{\beta}}'(t) = \Omega_{ConeWt}\, \dot{\underline{\beta}}_{Cnt}(t) \tag{8.2.1.1-11}$$

with

$$\dot{\underline{\beta}}_{Cnt}(t) = \frac{1}{2}\left(\underline{\alpha}_{Cnt}(t) \times \underline{\omega}_{Puls}\right) \qquad \underline{\alpha}_{Cnt} = \int_{t_{m-1}}^{t} \underline{\omega}_{Puls}\, d\tau \tag{8.2.1.1-12}$$

$$\Omega_{ConeWt} = \begin{bmatrix} \Omega_{Wt_Y}\,\Omega_{Wt_Z} & 0 & 0 \\ 0 & \Omega_{Wt_Z}\,\Omega_{Wt_X} & 0 \\ 0 & 0 & \Omega_{Wt_X}\,\Omega_{Wt_Y} \end{bmatrix} \tag{8.2.1.1-13}$$

where

$\dot{\underline{\beta}}_{Cnt}(t) = \dot{\underline{\beta}}(t)$ computed with uncompensated inertial sensor output pulse data.

Ω_{Wt_i} = Element in row i, column i of Ω_{Wt}.

The integral of Equations (8.2.1.1-7) and (8.2.1.1-11) over an attitude update cycle defines the algorithm for compensating the coning term calculated from uncompensated angular rate sensor output data:

$$\underline{\beta}_{Cnt_m} = \frac{1}{2}\int_{t_{m-1}}^{t_m} \underline{\alpha}_{Cnt}(t) \times d\underline{\alpha}_{Cnt} \qquad \underline{\alpha}_{Cnt}(t) = \int_{t_{m-1}}^{t} d\underline{\alpha}_{Cnt}$$
$$\underline{\beta}'_m = \Omega_{ConeWt}\,\underline{\beta}_{Cnt_m} \qquad \underline{\beta}_m = \left(I - K_{MisCone}\right)\underline{\beta}'_m \tag{8.2.1.1-14}$$

where

$d\underline{\alpha}_{Cnt} = \underline{\omega}_{Puls}\,dt$ = Uncompensated angular rate sensor triad output differential pulse count vector.

The algorithmic implementation of Equation (8.2.1.1-14) would utilize the Equations (7.1.1.1.1-17) and (7.1.1.1.1-18) digital integration routine for the $\underline{\beta}_{Cnt_m}$ calculation exactly as $\underline{\beta}_m$ is calculated in (7.1.1.1.1-17) and (7.1.1.1.1-18), but using uncompensated angular rate data:

$$\Delta\underline{\alpha}_{Cnt_l} = \int_{t_{l-1}}^{t_l} d\underline{\alpha}_{Cnt} \qquad \text{Summation Increment Of Angular Rate Sensor Output Pulses}$$
$$\underline{\alpha}_{Cnt_l} = \underline{\alpha}_{Cnt_{l-1}} + \Delta\underline{\alpha}_{Cnt_l} \tag{8.2.1.1-15}$$
$$\underline{\alpha}_{Cnt_m} = \underline{\alpha}_{Cnt_l}(t_l = t_m) \qquad \underline{\alpha}_{Cnt_l} = 0 \;\text{ At }\; t = t_{m-1}.$$

$$\Delta\underline{\beta}_{Cnt_l} = \frac{1}{2}\left(\underline{\alpha}_{Cnt_{l-1}} + \frac{1}{6}\,\Delta\underline{\alpha}_{Cnt_{l-1}}\right) \times \Delta\underline{\alpha}_{Cnt_l}$$
$$\underline{\beta}_{Cnt_l} = \underline{\beta}_{Cnt_{l-1}} + \Delta\underline{\beta}_{Cnt_l} \tag{8.2.1.1-16}$$
$$\underline{\beta}_{Cnt_m} = \underline{\beta}_{Cnt_l}(t_l = t_m) \qquad \underline{\beta}_{Cnt_l} = 0 \;\text{ At }\; t = t_{m-1}.$$

$$\underline{\beta}'_m = \Omega_{ConeWt}\,\underline{\beta}_{Cnt_m} \qquad \underline{\beta}_m = \left(I - K_{MisCone}\right)\underline{\beta}'_m \tag{8.2.1.1-17}$$

It should be noted that in some applications, sufficient accuracy may be obtainable by recognizing that $K_{MisCone}$ is small and potentially negligible in calculating $\underline{\beta}_m$. In this case the approximation can be made that $\underline{\beta}_m$ equals $\underline{\beta}'_m$ in Equations (8.2.1.1-17).

Equations (8.2.1.1-15) - (8.2.1.1-17) for $\underline{\beta}'_m$ are based on the assumption that Ω_{Wt} can be approximated as constant over an m cycle. Under this assumption, if Ω_{Wt} contains non-linear terms (as addressed in Section 8.1.1.3), Ω_{Wt} would be calculated as a function of $\underline{\alpha}_{Cnt_m}$. For the more general case in which Ω_{Wt} may have rapid variations, $\underline{\beta}'_m$ can be calculated based on Equation (8.2.1-1), but computed from scale factor compensated angular rate sensor output data $\underline{\omega}'$ as in (8.2.1.1-1). Then the equivalent to (8.2.1.1-15) - (8.2.1.1-17) for $\underline{\beta}_m$ becomes:

$$\Delta\underline{\alpha}_{Cnt_l} = \int_{t_{l-1}}^{t_l} d\underline{\alpha}_{Cnt} \qquad \text{Summation Increment Of Angular Rate Sensor Output Pulses}$$

$$\Delta\underline{\alpha}'_l = \Omega_{Wt}\,\Delta\underline{\alpha}_{Cnt_l} \tag{8.2.1.1-18}$$

$$\underline{\alpha}'_l = \underline{\alpha}'_{l-1} + \Delta\underline{\alpha}'_l$$

$$\underline{\alpha}'_m = \underline{\alpha}'_l(t_l = t_m) \qquad \underline{\alpha}'_l = 0 \quad \text{At } t = t_{m-1}.$$

$$\Delta\underline{\beta}'_l = \frac{1}{2}\left(\underline{\alpha}'_{l-1} + \frac{1}{6}\,\Delta\underline{\alpha}'_{l-1}\right) \times \Delta\underline{\alpha}'_l$$

$$\underline{\beta}'_l = \underline{\beta}'_{l-1} + \Delta\underline{\beta}'_l \tag{8.2.1.1-19}$$

$$\underline{\beta}'_m = \underline{\beta}'_l(t_l = t_m) \qquad \underline{\beta}'_l = 0 \quad \text{At } t = t_{m-1}.$$

$$\underline{\beta}_m = \left(I - K_{MisCone}\right)\underline{\beta}'_m \tag{8.2.1.1-20}$$

Finally, if angular rate sensor scale factor asymmetry compensation is being applied in the software algorithms as described in Section 8.1.1.3, Equation (8.2.1.1-18) would be based on $\underline{\omega}'$ given by Equation (8.1.1.3-21) rather than (8.2.1.1-1):

$$\underline{\omega}' = \Omega_{Wt_+}\,\underline{\omega}_{+Puls} + \Omega_{Wt_-}\,\underline{\omega}_{-Puls} \tag{8.2.1.1-21}$$

The $\Omega_{Wt\,+}$, $\Omega_{Wt\,-}$ terms in Equation (8.2.1.1-21) are scale factor weighting matrices defined in Equation (8.1.1.3-22) for positive and negative uncompensated angular rate sensor pulse rate output data ($\underline{\omega}_{+Puls}$ and $\underline{\omega}_{-Puls}$ - See Section 8.1.1.3 for further clarification). Based on Equation (8.2.1.1-21), the equivalent to the (8.2.1.1-18) operations would be as follows:

$$\Delta\underline{\alpha}_{+Cnt_l} \equiv \int_{t_{l-1}}^{t_l} d\underline{\alpha}_{+Cnt} \qquad \text{Summation Increment Of Positive Angular Rate Sensor Output Pulses}$$

$$\Delta\underline{\alpha}_{-Cnt_l} \equiv \int_{t_{l-1}}^{t_l} d\underline{\alpha}_{-Cnt} \qquad \text{Summation Increment Of Negative Angular Rate Sensor Output Pulses}$$

$$\Delta\underline{\alpha}'_l = \Omega_{Wt_+}\,\Delta\underline{\alpha}_{+Cnt_l} + \Omega_{Wt_-}\,\Delta\underline{\alpha}_{-Cnt_l} \tag{8.2.1.1-22}$$

$$\underline{\alpha}'_l = \underline{\alpha}'_{l-1} + \Delta\underline{\alpha}'_l$$

$$\underline{\alpha}'_m = \underline{\alpha}'_l(t_l = t_m) \qquad \underline{\alpha}'_l = 0 \quad \text{At } t = t_{m-1}.$$

where

$d\underline{\alpha}_{+Cnt}$, $d\underline{\alpha}_{-Cnt}$ = $\underline{\omega}_{+Puls}$ dt and $\underline{\omega}_{-Puls}$ dt, the uncompensated angular rate differential positive and negative pulse rate outputs.

Equations (8.2.1.1-22) are based on the assumption that separate plus and minus pulse sum outputs are provided from the angular rate sensor interface to the navigation software. If only a composite pulse sum is available (as in (8.2.1.1-18)), Equations (8.2.1.1-22) would be of the form:

$$\Delta\underline{\alpha}_{Cnt_l} = \int_{t_{l-1}}^{t_l} d\underline{\alpha}_{Cnt} \qquad \text{Summation Increment Of Angular Rate Sensor Output Pulses}$$

Do For Each Component Of $\Delta\underline{\alpha}_{Cnt_l}$:

$$\text{If } \Delta\alpha_{iCnt_l} \geq 0 \quad \text{Then:} \quad \Delta\alpha'_{il} = \Omega_{Wt_{i+}}\,\Delta\alpha_{iCnt_l} \tag{8.2.1.1-23}$$

$$\text{Else:} \quad \Delta\alpha'_{il} = \Omega_{Wt_{i-}}\,\Delta\alpha_{iCnt_l}$$

$$\underline{\alpha}'_l = \underline{\alpha}'_{l-1} + \Delta\underline{\alpha}'_l$$

$$\underline{\alpha}'_m = \underline{\alpha}'_l(t_l = t_m) \qquad \underline{\alpha}'_l = 0 \quad \text{At } t = t_{m-1}.$$

where

$\Delta\alpha_{iCnt_l}$, $\Delta\alpha'_{il}$ = Components i of $\underline{\alpha}_{Cnt_l}$, $\Delta\underline{\alpha}'_l$.

$\Omega_{Wt_{i+}}$, $\Omega_{Wt_{i-}}$ = Elements i, i in Ω_{Wt_+}, Ω_{Wt_-}.

8.2.2 INERTIAL SENSOR COMPENSATION FOR VELOCITY UPDATING

The accelerometer and angular rate sensor outputs in a strapdown system are used through suitable algorithms (Section 7.2) to update the system velocity data. In general, this is achieved by calculating an integrated specific force acceleration increment in body B Frame coordinates with Equations (7.2.2.2-23), (7.2.2.2-24) and (7.2.2.2-25) (or (7.2.2.2.1-7) as an option) or Equations (7.2.2.2-26) - (7.2.2.2-27), all repeated below, over an acceleration transformation update cycle. The integrated acceleration increment is then used to update the velocity data. The sculling vector obtained in these calculations is also identical to the sculling portion of the velocity translation vector (analogous to the rotation vector) of Section 19.1, Equations (19.1.8-3) for updating velocity in Equations (19.1.5-9) based on a new unified strapdown algorithm concept. The sculling portion of the velocity translation vector is the integral of $\dot{\underline{\eta}}_{Algo/c}$ in (19.1.8-3) without the $\underline{a}_{SF}$ term.

$$\Delta\underline{v}_{SF_m} = \underline{\upsilon}_m + \Delta\underline{v}_{Rot_m} + \Delta\underline{v}_{Scul_m}$$

$$\Delta\underline{v}_{Rot_m} = \frac{1}{2}\left(\underline{\alpha}_m \times \underline{\upsilon}_m\right) \qquad \text{or Equation (7.2.2.2.1-7) as option}$$

$$\Delta\underline{v}_{Scul_m} = \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{a}_{SF} + \underline{\upsilon}(t) \times \underline{\omega}\right) dt \tag{8.2.2-1}$$

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}\, d\tau \qquad \underline{\alpha}_m = \underline{\alpha}(t_m)$$

$$\underline{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{a}_{SF}\, d\tau \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)$$

or

$$\Delta\underline{v}_{SF_m} = \underline{\upsilon}_m + \Delta\underline{v}_{Rot/Scul_m}$$

$$\Delta\underline{v}_{Rot/Scul_m} = \int_{t_{m-1}}^{t_m} \left(\underline{\alpha}(t) \times \underline{a}_{SF}\right) dt$$

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}\, d\tau \tag{8.2.2-2}$$

$$\underline{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{a}_{SF}\, d\tau \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)$$

where

$\Delta\underline{v}_{SF_m}$ = Integrated specific force acceleration increment.

$\underline{a}_{SF}$ = Specific force acceleration vector sensed by accelerometer triad.

$\underline{\omega}$ = Angular rate vector sensed by angular rate sensor triad.

m = Acceleration-transformation/velocity-update cycle index.

$\underline{\upsilon}_m$ = Integrated specific force vector from t_{m-1} to t_m.

$\underline{\alpha}_m$ = Integrated angular rate vector from t_{m-1} to t_m.

$\Delta\underline{v}_{Rot_m}$ = "Velocity Rotation Compensation" term.

$\Delta\underline{v}_{Scul_m}$ = "Sculling" term.

$\Delta\underline{v}_{Rot/Scul_m}$ = Composite "Sculling" and "Velocity Rotation Compensation" term.

All vectors in (8.2.2-1) and (8.2.2-2) are in B Frame coordinates, the superscript notation for which has been omitted for simplicity. Similarly, the IB subscript has been dropped from the $\underline{\omega}$ angular rate sensor vector which was identified in Chapter 7 as $\underline{\omega}_{IB}$.

Equations (8.2.2-1) and (8.2.2-2) are based on the use of compensated accelerometer and angular rate sensor data. Equations (8.1.2.1-3) (or (8.1.2.1-7)), (8.1.2.1-4), (8.1.2.2-3) (or (8.1.2.2-7)) and (8.1.2.2-4) provide the means for calculating the compensated form of $\underline{\alpha}_m$ and $\underline{\upsilon}_m$ for Equations (8.2.2-1) and (8.2.2-2). The $\Delta\underline{v}_{Scul_m}$ sculling term in (8.2.2-1) and the $\Delta\underline{v}_{Rot/Scul_m}$ composite sculling/rotation-compensation term in (8.2.2-2) can be calculated using compensated $\underline{\omega}$, $\underline{a}_{SF}$ angular-rate/specific-force input data based on compensation Equations (8.1.2.1-2) and (8.1.2.2-2). A problem with this approach is that the $\underline{\omega}$, $\underline{a}_{SF}$ compensation operations would have to be performed at the high rate used to calculate $\Delta\underline{v}_{Scul_m}$ or $\Delta\underline{v}_{Rot/Scul_m}$ (i.e., the *l* computer cycle rate as depicted in Equations (7.2.2.2.2-15) or (7.2.2.2.2-24) which are the computer implementations of Equations (8.2.2-1) and (8.2.2-2)). Alternatively, the Equations (7.2.2.2.2-15) sculling algorithm or the Equations (7.2.2.2.2-24) composite sculling/rotation-compensation algorithm can be calculated with uncompensated angular rate and accelerometer sensor data, with the uncompensated sculling or sculling/rotation-compensation solutions then compensated for sensor error at the acceleration-transformation/velocity-update rate (i.e., the m rate in Equations (7.2.2.2.2-15) or (7.2.2.2.2-24)). The following sections describe the latter approach.

8.2.2.1 SCULLING INCREMENT COMPENSATION ALGORITHM

Development of the algorithm for compensating the Equation (8.2.2-1) $\Delta\underline{v}_{Scul_m}$ sculling term begins with an approximate form of Equations (8.1.2.1-2) and (8.1.2.2-2) which neglects the quantization terms as second order for sculling increment compensation purposes:

$$\underline{\omega}' = \Omega_{Wt}\, \underline{\omega}_{Puls} \tag{8.2.2.1-1}$$

$$\underline{\omega} \approx \underline{\omega}' - K_{Mis}\, \underline{\omega}' - \underline{K}_{Bias} \tag{8.2.2.1-2}$$

$$\underline{a}_{SF}' = A_{Wt}\, \underline{a}_{SF_{Puls}} \tag{8.2.2.1-3}$$

$$\underline{a}_{SF} \approx \underline{a}_{SF}' - L_{Mis}\, \underline{a}_{SF}' - \underline{L}_{Bias}^{*} \tag{8.2.2.1-4}$$

with

$$\underline{L}_{Bias}^{*} \equiv \underline{L}_{Bias} + G_{Algn}^{-1} \left(\delta\underline{a}_{Size} + \delta\underline{a}_{Aniso} \right) \tag{8.2.2.1-5}$$

where

$\underline{\omega}_{Puls}$ = Uncompensated angular rate sensor output pulse rate vector (pulses per second).

Ω_{Wt} = Diagonal angular rate sensor pulse weighting matrix (radians per pulse) defined by Equation (8.1.1.1-5).

K_{Mis}, $\underline{K}_{Bias}$ = Angular rate sensor triad misalignment matrix and bias vector compensation coefficients as defined in Equations (8.1.1.1-6) - (8.1.1.1-7).

$\underline{a}_{SF_{Puls}}$ = Uncompensated accelerometer output pulse rate vector (pulses per second).

A_{Wt} = Diagonal accelerometer pulse weighting matrix (feet per second per pulse) defined by Equation (8.1.1.2-5).

L_{Mis}, $\underline{L}_{Bias}$ = Accelerometer triad misalignment matrix and bias vector compensation coefficients as defined in Equations (8.1.1.2-6) - (8.1.1.2-7).

G_{Align}, $\delta\underline{a}_{Size}$, $\delta\underline{a}_{Aniso}$ = Accelerometer triad alignment matrix, size effect vector and anisoinertia vector as defined following Equations (8.1.1.2-1).

$\underline{L}_{Bias}^{*}$ = Accelerometer triad equivalent bias vector.

Substituting (8.2.2.1-1) - (8.2.2.1-4) into the (8.2.2-1) $\underline{\alpha}(t)$, $\underline{\upsilon}(t)$ expressions obtains:

$$\underline{\alpha}'(t) = \Omega_{Wt}\, \underline{\alpha}_{Cnt}(t) \tag{8.2.2.1-6}$$

$$\underline{\alpha}_{Cnt}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{Puls}\, d\tau \tag{8.2.2.1-7}$$

$$\underline{\alpha}(t) \approx \underline{\alpha}'(t) - K_{Mis}\, \underline{\alpha}'(t) - \underline{K}_{Bias} \left(t - t_{m-1} \right) \tag{8.2.2.1-8}$$

$$\underline{\upsilon}'(t) = A_{Wt}\, \underline{\upsilon}_{Cnt}(t) \tag{8.2.2.1-9}$$

$$\underline{\upsilon}_{Cnt}(t) \;=\; \int_{t_{m-1}}^{t} \underline{a}_{SF_{Puls}} \, d\tau \tag{8.2.2.1-10}$$

$$\underline{\upsilon}(t) \;\approx\; \underline{\upsilon}'(t) - L_{Mis}\,\underline{\upsilon}'(t) - \underline{L}^{*}_{Bias}\left(t - t_{m-1}\right) \tag{8.2.2.1-11}$$

where

$\underline{\alpha}_{Cnt}(t),\ \underline{\upsilon}_{Cnt}(t)$ = $\underline{\alpha}(t),\ \underline{\upsilon}(t)$ computed from uncompensated angular rate sensor and accelerometer triad output data.

We now substitute (8.2.2.1-2), (8.2.2.1-4), (8.2.2.1-8) and (8.2.2.1-11) into the (8.2.2-1) $\Delta\underline{v}_{Scul_m}$ expression to obtain for the integrand:

$$\begin{aligned}
\Delta\dot{\underline{v}}_{Scul}(t) \;&=\; \frac{1}{2}\left(\underline{\alpha}(t)\times\underline{a}_{SF} + \underline{\upsilon}(t)\times\underline{\omega}\right) \;=\; \frac{1}{2}\left(\underline{\alpha}(t)\times\underline{a}_{SF} - \underline{\omega}\times\underline{\upsilon}(t)\right) \\
&= \frac{1}{2}\Big\{\left[\underline{\alpha}'(t) - K_{Mis}\,\underline{\alpha}'(t) - \underline{K}_{Bias}\left(t - t_{m-1}\right)\right]\times\left[\underline{a}'_{SF} - L_{Mis}\,\underline{a}'_{SF} - \underline{L}^{*}_{Bias}\right] \\
&\qquad - \left[\underline{\omega}' - K_{Mis}\,\underline{\omega}' - \underline{K}_{Bias}\right]\times\left[\underline{\upsilon}'(t) - L_{Mis}\,\underline{\upsilon}'(t) - \underline{L}^{*}_{Bias}\left(t - t_{m-1}\right)\right] \\
&\approx \frac{1}{2}\Big\{\left[\underline{\alpha}'(t)\times\underline{a}'_{SF} - \underline{\omega}'\times\underline{\upsilon}'(t)\right] - \left[\left(K_{Mis}\,\underline{\alpha}'(t)\right)\times\underline{a}'_{SF} - \left(K_{Mis}\,\underline{\omega}'\right)\times\underline{\upsilon}'(t)\right] \\
&\qquad - \left[\underline{\alpha}'(t)\times\left(L_{Mis}\,\underline{a}'_{SF}\right) - \underline{\omega}'\times\left(L_{Mis}\,\underline{\upsilon}'(t)\right)\right]\Big\} \\
&\qquad - \underline{K}_{Bias}\times\left[\underline{a}'_{SF}\left(t - t_{m-1}\right) - \underline{\upsilon}'(t)\right] - \left[\underline{\alpha}'(t) - \underline{\omega}'\left(t - t_{m-1}\right)\right]\times\underline{L}^{*}_{Bias} \\
&\approx \frac{1}{2}\Big\{\left[\underline{\alpha}'(t)\times\underline{a}'_{SF} - \underline{\omega}'\times\underline{\upsilon}'(t)\right] - \left[\left(K_{Mis}\,\underline{\alpha}'(t)\right)\times\underline{a}'_{SF} - \left(K_{Mis}\,\underline{\omega}'\right)\times\underline{\upsilon}'(t)\right] \\
&\qquad - \left[\underline{\alpha}'(t)\times\left(L_{Mis}\,\underline{a}'_{SF}\right) - \underline{\omega}'\times\left(L_{Mis}\,\underline{\upsilon}'(t)\right)\right]\Big\} \\
&= \Delta\dot{\underline{v}}'_{Scul}(t) - \frac{1}{2}\Big\{\left[\left(K_{Mis}\,\underline{\alpha}'(t)\right)\times\underline{a}'_{SF} - \left(K_{Mis}\,\underline{\omega}'\right)\times\underline{\upsilon}'(t)\right] \\
&\qquad + \left[\underline{\alpha}'(t)\times\left(L_{Mis}\,\underline{a}'_{SF}\right) - \underline{\omega}'\times\left(L_{Mis}\,\underline{\upsilon}'(t)\right)\right]\Big\}
\end{aligned} \tag{8.2.2.1-12}$$

with

$$\Delta\dot{\underline{v}}'_{Scul}(t) \;\equiv\; \frac{1}{2}\left(\underline{\alpha}'(t)\times\underline{a}'_{SF} - \underline{\omega}'\times\underline{\upsilon}'(t)\right) \tag{8.2.2.1-13}$$

where

$\Delta\dot{\underline{v}}'_{Scul}(t)$ = $\Delta\dot{\underline{v}}_{Scul}(t)$ computed with scaled but uncompensated inertial sensor data (except for scale factor error correction which is included in the scaling operation).

Equation (8.2.2.1-12) is based on dropping K_{Mis}, L_{Mis}, $\underline{K}_{Bias}$, $\underline{L}^*_{Bias}$ product terms as second order, approximating $\underline{\omega}'$ as constant in the $\left[\underline{\alpha}'(t) - \underline{\omega}'\left(t - t_{m-1}\right)\right] \times \underline{L}^*_{Bias}$ expression so that $\underline{\alpha}'(t) \approx \underline{\omega}'\,(t - t_{m-1})$, and approximating $\underline{a}'_{SF}$ as constant in the $\underline{K}_{Bias} \times \left[\underline{a}'_{SF}\left(t - t_{m-1}\right) - \underline{\upsilon}'(t)\right]$ expression so that $\underline{\upsilon}'(t) \approx \underline{a}'_{SF}\left(t - t_{m-1}\right)$.

Further development of the sculling compensation algorithm is facilitated by first defining a more basic vector product terminology. The cross-product of two arbitrary vectors is given by:

$$\underline{V} = \underline{A} \times \underline{B} = \begin{bmatrix} A_Y\, B_Z - A_Z\, B_Y \\ A_Z\, B_X - A_X\, B_Z \\ A_X\, B_Y - A_Y\, B_X \end{bmatrix} \tag{8.2.2.1-14}$$

where

$\underline{A}$, $\underline{B}$ = Arbitrary vectors.

$\underline{V}$ = Cross-product between $\underline{A}$ and $\underline{B}$.

A_i, B_i = Components of $\underline{A}$, $\underline{B}$.

We define the following vector product operators:

$$\underline{V}_1 \equiv \begin{bmatrix} A_Y\, B_Z \\ A_Z\, B_X \\ A_X\, B_Y \end{bmatrix} \qquad \underline{V}_2 \equiv \begin{bmatrix} A_Z\, B_Y \\ A_X\, B_Z \\ A_Y\, B_X \end{bmatrix} \qquad \underline{V}_3 \equiv \begin{bmatrix} A_X\, B_X \\ A_Y\, B_Y \\ A_Z\, B_Z \end{bmatrix} \tag{8.2.2.1-15}$$

Using the (8.2.2.1-15) definitions in (8.2.2.1-14), the $\underline{V}$ cross-product vector is given by:

$$\underline{V} = \underline{V}_1 - \underline{V}_2 \tag{8.2.2.1-16}$$

Now consider a vector $\underline{W}$ formed as a sum of cross-product terms:

$$\underline{W} = \underline{V}_{AB} + \underline{V}_{CD} + \underline{V}_{EF} + \cdots \tag{8.2.2.1-17}$$

with

$$\underline{V}_{AB} \equiv \underline{A} \times \underline{B} \qquad \underline{V}_{CD} \equiv \underline{C} \times \underline{D} \qquad \underline{V}_{EF} \equiv \underline{E} \times \underline{F} \tag{8.2.2.1-18}$$

where

$\underline{A}$, $\underline{B}$, $\underline{C}$, $\underline{D}$, $\underline{E}$, $\underline{F}$ = Arbitrary vectors.

Applying the Equation (8.2.2.1-15) definitions to Equation (8.2.2.1-17) obtains:

$$\underline{W}_1 = \underline{V}_{AB_1} + \underline{V}_{CD_1} + \underline{V}_{EF_1} + \cdots \tag{8.2.2.1-19}$$

$$\underline{W}_2 = \underline{V}_{AB_2} + \underline{V}_{CD_2} + \underline{V}_{EF_2} + \cdots \tag{8.2.2.1-20}$$

$$\underline{W}_3 = \underline{V}_{AB_3} + \underline{V}_{CD_3} + \underline{V}_{EF_3} + \cdots \tag{8.2.2.1-21}$$

where

$(\ )_1, (\ )_2, (\ )_3$ = Vector component product operators as defined by Equations (8.2.2.1-15).

Then, the vector $\underline{W}$ in (8.2.2.1-17) formed as the sum of cross-products becomes:

$$\underline{W} = \underline{W}_1 - \underline{W}_2 \tag{8.2.2.1-22}$$

Let us now use the Equation (8.2.2.1-15) operators with the (8.2.2.1-19) - (8.2.2.1-21) rules to $\Delta\dot{\underline{v}}'_{Scul}(t)$ in Equation (8.2.2.1-13) to define the following $\Delta\dot{\underline{v}}'_{Scul_1}(t)$, $\Delta\dot{\underline{v}}'_{Scul_2}(t)$, $\Delta\dot{\underline{v}}'_{Scul_3}(t)$ component parameters:

$$\Delta\dot{\underline{v}}'_{Scul_1}(t) = \frac{1}{2}\begin{bmatrix} \alpha'_Y(t)\, a'_{SF_Z} - \upsilon'_Z(t)\, \omega'_Y \\ \alpha'_Z(t)\, a'_{SF_X} - \upsilon'_X(t)\, \omega'_Z \\ \alpha'_X(t)\, a'_{SF_Y} - \upsilon'_Y(t)\, \omega'_X \end{bmatrix}$$

$$\Delta\dot{\underline{v}}'_{Scul_2}(t) = \frac{1}{2}\begin{bmatrix} \alpha'_Z(t)\, a'_{SF_Y} - \upsilon'_Y(t)\, \omega'_Z \\ \alpha'_X(t)\, a'_{SF_Z} - \upsilon'_Z(t)\, \omega'_X \\ \alpha'_Y(t)\, a'_{SF_X} - \upsilon'_X(t)\, \omega'_Y \end{bmatrix} \tag{8.2.2.1-23}$$

$$\Delta\dot{\underline{v}}'_{Scul_3}(t) = \frac{1}{2}\begin{bmatrix} \alpha'_X(t)\, a'_{SF_X} - \upsilon'_X(t)\, \omega'_X \\ \alpha'_Y(t)\, a'_{SF_Y} - \upsilon'_Y(t)\, \omega'_Y \\ \alpha'_Z(t)\, a'_{SF_Z} - \upsilon'_Z(t)\, \omega'_Z \end{bmatrix}$$

where

$\upsilon'_X(t)\,,\ \upsilon'_Y(t)\,,\ \upsilon'_Z(t)$ = Components of $\underline{\upsilon}'(t)$.

$a'_{SF_X}, a'_{SF_Y}, a'_{SF_Z}$ = Components of $\underline{a}'_{SF}$.

$\alpha'_X(t)\,,\ \alpha'_Y(t)\,,\ \alpha'_Z(t)$ = Components of $\underline{\alpha}'(t)$.

$\omega'_X, \omega'_Y, \omega'_Z$ = Components of $\underline{\omega}'$.

We also define the equivalent version of Equation (8.2.2.1-13) based on uncompensated angular rate sensor and accelerometer output data:

$$\Delta \dot{\underline{v}}_{SculCnt}(t) \equiv \frac{1}{2}\left(\underline{\alpha}_{Cnt}(t) \times \underline{a}_{SF_{Puls}} - \underline{\omega}_{Puls} \times \underline{\upsilon}_{Cnt}(t)\right) \qquad (8.2.2.1\text{-}24)$$

where

$\Delta \dot{\underline{v}}_{SculCnt}(t) = \Delta \dot{\underline{v}}_{Scul}(t)$ calculated with uncompensated inertial sensor output pulse data.

Applying the Equation (8.2.2.1-15) operators with the (8.2.2.1-19) - (8.2.2.1-21) rules to $\Delta \dot{\underline{v}}_{SculCnt}(t)$ in Equation (8.2.2.1-24) defines the following $\Delta \dot{\underline{v}}_{SculCnt_1}(t)$, $\Delta \dot{\underline{v}}_{SculCnt_2}(t)$, $\Delta \dot{\underline{v}}_{SculCnt_3}(t)$ version vector parameters:

$$\Delta \dot{\underline{v}}_{SculCnt_1}(t) = \frac{1}{2}\begin{bmatrix} \alpha_{Cnt_Y}(t)\ a_{SFPuls_Z} - \upsilon_{Cnt_Z}(t)\ \omega_{Puls_Y} \\ \alpha_{Cnt_Z}(t)\ a_{SFPuls_X} - \upsilon_{Cnt_X}(t)\ \omega_{Puls_Z} \\ \alpha_{Cnt_X}(t)\ a_{SFPuls_Y} - \upsilon_{Cnt_Y}(t)\ \omega_{Puls_X} \end{bmatrix}$$

$$\Delta \dot{\underline{v}}_{SculCnt_2}(t) = \frac{1}{2}\begin{bmatrix} \alpha_{Cnt_Z}(t)\ a_{SFPuls_Y} - \upsilon_{Cnt_Y}(t)\ \omega_{Puls_Z} \\ \alpha_{Cnt_X}(t)\ a_{SFPuls_Z} - \upsilon_{Cnt_Z}(t)\ \omega_{Puls_X} \\ \alpha_{Cnt_Y}(t)\ a_{SFPuls_X} - \upsilon_{Cnt_X}(t)\ \omega_{Puls_Y} \end{bmatrix} \qquad (8.2.2.1\text{-}25)$$

$$\Delta \dot{\underline{v}}_{SculCnt_3}(t) = \frac{1}{2}\begin{bmatrix} \alpha_{Cnt_X}(t)\ a_{SFPuls_X} - \upsilon_{Cnt_X}(t)\ \omega_{Puls_X} \\ \alpha_{Cnt_Y}(t)\ a_{SFPuls_Y} - \upsilon_{Cnt_Y}(t)\ \omega_{Puls_Y} \\ \alpha_{Cnt_Z}(t)\ a_{SFPuls_Z} - \upsilon_{Cnt_Z}(t)\ \omega_{Puls_Z} \end{bmatrix}$$

where

$\alpha_{Cnt_X}(t),\ \alpha_{Cnt_Y}(t),\ \alpha_{Cnt_Z}(t)$ = Components of $\underline{\alpha}_{Cnt}(t)$.

$\omega_{Puls_X}, \omega_{Puls_Y}, \omega_{Puls_Z}$ = Components of $\underline{\omega}_{Puls}$.

$\upsilon_{Cnt_X}(t),\ \upsilon_{Cnt_Y}(t),\ \upsilon_{Cnt_Z}(t)$ = Components of $\underline{\upsilon}_{Cnt}(t)$.

$a_{SFPuls_X}, a_{SFPuls_Y}, a_{SFPuls_Z}$ = Components of $\underline{a}_{SF_{Puls}}$.

From the forms of Equations (8.2.2.1-23) and (8.2.2.1-25), we see with Equations (8.2.2.1-1), (8.2.2.1-3), (8.2.2.1-6) and (8.2.2.1-9) that:

$$\Delta \dot{\underline{v}}'_{Scul_1}(t) = \Omega A_{Wt_1} \Delta \dot{\underline{v}}_{SculCnt_1}(t)$$
$$\Delta \dot{\underline{v}}'_{Scul_2}(t) = \Omega A_{Wt_2} \Delta \dot{\underline{v}}_{SculCnt_2}(t) \qquad (8.2.2.1\text{-}26)$$
$$\Delta \dot{\underline{v}}'_{Scul_3}(t) = \Omega A_{Wt_3} \Delta \dot{\underline{v}}_{SculCnt_3}(t)$$

with

$$\Omega A_{Wt_1} = \begin{bmatrix} \Omega_{Wt_Y} A_{Wt_Z} & 0 & 0 \\ 0 & \Omega_{Wt_Z} A_{Wt_X} & 0 \\ 0 & 0 & \Omega_{Wt_X} A_{Wt_Y} \end{bmatrix}$$

$$\Omega A_{Wt_2} = \begin{bmatrix} \Omega_{Wt_Z} A_{Wt_Y} & 0 & 0 \\ 0 & \Omega_{Wt_X} A_{Wt_Z} & 0 \\ 0 & 0 & \Omega_{Wt_Y} A_{Wt_X} \end{bmatrix} \qquad (8.2.2.1\text{-}27)$$

$$\Omega A_{Wt_3} = \begin{bmatrix} \Omega_{Wt_X} A_{Wt_X} & 0 & 0 \\ 0 & \Omega_{Wt_Y} A_{Wt_Y} & 0 \\ 0 & 0 & \Omega_{Wt_Z} A_{Wt_Z} \end{bmatrix}$$

where

Ω_{Wt_i}, A_{Wt_i} = Elements in row i, column i of Ω_{Wt}, A_{Wt}.

With Equation (8.2.2.1-22) we can utilize the Equations (8.2.2.1-26) results to calculate $\Delta \dot{\underline{v}}'_{Scul}(t)$ for Equation (8.2.2.1-12):

$$\Delta \dot{\underline{v}}'_{Scul}(t) = \Delta \dot{\underline{v}}'_{Scul_1}(t) - \Delta \dot{\underline{v}}'_{Scul_2}(t)) \qquad (8.2.2.1\text{-}28)$$

It will be beneficial to also define:

$$\Delta \dot{\underline{v}}'_{SculR}(t) \equiv \Delta \dot{\underline{v}}'_{Scul_1}(t) + \Delta \dot{\underline{v}}'_{Scul_2}(t)$$
$$\Delta \dot{\underline{v}}'_{SculQ} \equiv \Delta \dot{\underline{v}}'_{Scul_3}(t) \qquad (8.2.2.1\text{-}29)$$

We now expand the individual elements of $\Delta\dot{\underline{v}}_{Scul}(t)$ Equation (8.2.2.1-12) in scalar component form for all but the $\Delta\dot{\underline{v}}'_{Scul}(t)$ term. After collection and rearrangement of the detailed results, regrouping of terms, and substitution of Equations (8.2.2.1-23), (8.2.2.1-25), (8.2.2.1-28) and (8.2.2.1-29), the following is obtained:

$$\Delta\dot{\underline{v}}_{Scul}(t) \;=\; \Delta\dot{\underline{v}}'_{Scul}(t) - LK_1\, \Delta\dot{\underline{v}}'_{Scul}(t) - LK_2\, \Delta\dot{\underline{v}}'_{SculR}(t) - LK_3\, \Delta\dot{\underline{v}}'_{SculQ}(t) \qquad (8.2.2.1\text{-}30)$$

in which:

$$LK_1 \equiv \frac{1}{2}\begin{bmatrix} \begin{gathered}\left(L_{Mis_{YY}} + L_{Mis_{ZZ}}\right.\\ \left. + K_{Mis_{YY}} + K_{Mis_{ZZ}}\right)\end{gathered} & -\left(L_{Mis_{YX}} + K_{Mis_{YX}}\right) & -\left(L_{Mis_{ZX}} + K_{Mis_{ZX}}\right) \\ -\left(L_{Mis_{XY}} + K_{Mis_{XY}}\right) & \begin{gathered}\left(L_{Mis_{ZZ}} + L_{Mis_{XX}}\right.\\ \left. + K_{Mis_{ZZ}} + K_{Mis_{XX}}\right)\end{gathered} & -\left(L_{Mis_{ZY}} + K_{Mis_{ZY}}\right) \\ -\left(L_{Mis_{XZ}} + K_{Mis_{XZ}}\right) & -\left(L_{Mis_{YZ}} + K_{Mis_{YZ}}\right) & \begin{gathered}\left(L_{Mis_{XX}} + L_{Mis_{YY}}\right.\\ \left. + K_{Mis_{XX}} + K_{Mis_{YY}}\right)\end{gathered} \end{bmatrix}$$

$$LK_2 \equiv \frac{1}{2}\begin{bmatrix} \begin{gathered}\left(L_{Mis_{ZZ}} - L_{Mis_{YY}}\right.\\ \left. - K_{Mis_{ZZ}} + K_{Mis_{YY}}\right)\end{gathered} & -\left(L_{Mis_{YX}} - K_{Mis_{YX}}\right) & \left(L_{Mis_{ZX}} - K_{Mis_{ZX}}\right) \\ \left(L_{Mis_{XY}} - K_{Mis_{XY}}\right) & \begin{gathered}\left(L_{Mis_{XX}} - L_{Mis_{ZZ}}\right.\\ \left. - K_{Mis_{XX}} + K_{Mis_{ZZ}}\right)\end{gathered} & -\left(L_{Mis_{ZY}} - K_{Mis_{ZY}}\right) \\ -\left(L_{Mis_{XZ}} - K_{Mis_{XZ}}\right) & \left(L_{Mis_{YZ}} - K_{Mis_{YZ}}\right) & \begin{gathered}\left(L_{Mis_{YY}} - L_{Mis_{XX}}\right.\\ \left. - K_{Mis_{YY}} + K_{Mis_{XX}}\right)\end{gathered} \end{bmatrix} \qquad (8.2.2.1\text{-}31)$$

$$LK_3 \equiv \begin{bmatrix} 0 & \left(L_{Mis_{ZY}} - K_{Mis_{ZY}}\right) & -\left(L_{Mis_{YZ}} - K_{Mis_{YZ}}\right) \\ -\left(L_{Mis_{ZX}} - K_{Mis_{ZX}}\right) & 0 & \left(L_{Mis_{XZ}} - K_{Mis_{XZ}}\right) \\ \left(L_{Mis_{YX}} - K_{Mis_{YX}}\right) & -\left(L_{Mis_{XY}} - K_{Mis_{XY}}\right) & 0 \end{bmatrix}$$

where

$L_{Mis_{ij}}, K_{Mis_{ij}}$ = Elements i, j of L_{Mis}, K_{Mis}.

From the definition of K_{Mis} and L_{Mis} in Equations (8.1.1.1-6), (8.1.1.1.1-15), (8.1.1.1.1-16), (8.1.1.2-6), (8.1.1.2.1-13) and (8.1.1.2.1-14), it can be demonstrated that the diagonal elements are second order compared to the off-diagonal terms. This forms the basis for neglecting the K_{Mis} , L_{Mis} diagonal elements in Equations (8.2.2.1-31) to obtain the simpler versions of LK_1 and LK_2:

$$LK_1 \approx \frac{1}{2}\begin{bmatrix} 0 & -\left(L_{Mis_{YX}} + K_{Mis_{YX}}\right) & -\left(L_{Mis_{ZX}} + K_{Mis_{ZX}}\right) \\ -\left(L_{Mis_{XY}} + K_{Mis_{XY}}\right) & 0 & -\left(L_{Mis_{ZY}} + K_{Mis_{ZY}}\right) \\ -\left(L_{Mis_{XZ}} + K_{Mis_{XZ}}\right) & -\left(L_{Mis_{YZ}} + K_{Mis_{YZ}}\right) & 0 \end{bmatrix}$$

(8.2.2.1-32)

$$LK_2 \approx \frac{1}{2}\begin{bmatrix} 0 & -\left(L_{Mis_{YX}} - K_{Mis_{YX}}\right) & \left(L_{Mis_{ZX}} - K_{Mis_{ZX}}\right) \\ \left(L_{Mis_{XY}} - K_{Mis_{XY}}\right) & 0 & -\left(L_{Mis_{ZY}} - K_{Mis_{ZY}}\right) \\ -\left(L_{Mis_{XZ}} - K_{Mis_{XZ}}\right) & \left(L_{Mis_{YZ}} - K_{Mis_{YZ}}\right) & 0 \end{bmatrix}$$

Finally, the integral of Equations (8.2.2.1-25) - (8.2.2.1-26) and (8.2.2.1-28) - (8.2.2.1-30) over an acceleration-transformation/velocity-update cycle defines the algorithm for compensating the sculling term calculated from uncompensated inertial sensor output data:

$$d\Delta \underline{v}_{SculCnt_1} = \frac{1}{2}\begin{bmatrix} \alpha_{Cnt_Y}(t)\ d\upsilon_{Cnt_Z} - \upsilon_{Cnt_Z}(t)\ d\alpha_{Cnt_Y} \\ \alpha_{Cnt_Z}(t)\ d\upsilon_{Cnt_X} - \upsilon_{Cnt_X}(t)\ d\alpha_{Cnt_Z} \\ \alpha_{Cnt_X}(t)\ d\upsilon_{Cnt_Y} - \upsilon_{Cnt_Y}(t)\ d\alpha_{Cnt_X} \end{bmatrix}$$

$$d\Delta \underline{v}_{SculCnt_2} = \frac{1}{2}\begin{bmatrix} \alpha_{Cnt_Z}(t)\ d\upsilon_{Cnt_Y} - \upsilon_{Cnt_Y}(t)\ d\alpha_{Cnt_Z} \\ \alpha_{Cnt_X}(t)\ d\upsilon_{Cnt_Z} - \upsilon_{Cnt_Z}(t)\ d\alpha_{Cnt_X} \\ \alpha_{Cnt_Y}(t)\ d\upsilon_{Cnt_X} - \upsilon_{Cnt_X}(t)\ d\alpha_{Cnt_Y} \end{bmatrix}$$

$$d\Delta \underline{v}_{SculCnt_3} = \frac{1}{2}\begin{bmatrix} \alpha_{Cnt_X}(t)\ d\ \upsilon_{Cnt_X} - \upsilon_{Cnt_X}(t)\ d\ \alpha_{Cnt_X} \\ \alpha_{Cnt_Y}(t)\ d\ \upsilon_{Cnt_Y} - \upsilon_{Cnt_Y}(t)\ d\ \alpha_{Cnt_Y} \\ \alpha_{Cnt_Z}(t)\ d\ \upsilon_{Cnt_Z} - \upsilon_{Cnt_Z}(t)\ d\ \alpha_{Cnt_Z} \end{bmatrix} \qquad (8.2.2.1\text{-}33)$$

$$\Delta \underline{v}_{SculCnt_{1m}} = \int_{t_{m-1}}^{t_m} d\Delta \underline{v}_{SculCnt_1} \qquad \Delta \underline{v}_{SculCnt_{2m}} = \int_{t_{m-1}}^{t_m} d\Delta \underline{v}_{SculCnt_2}$$

$$\Delta \underline{v}_{SculCnt_{3m}} = \int_{t_{m-1}}^{t_m} d\Delta \underline{v}_{SculCnt_3}$$

$$\Delta \underline{v}'_{Scul_{1m}} = \Omega A_{Wt_1} \Delta \underline{v}_{SculCnt_{1m}} \qquad \Delta \underline{v}'_{Scul_{2m}} = \Omega A_{Wt_2} \Delta \underline{v}_{SculCnt_{2m}}$$

$$\Delta \underline{v}'_{Scul_{3m}} = \Omega A_{Wt_3} \Delta \underline{v}_{SculCnt_{3m}}$$

$$\Delta \underline{v}'_{Scul_m} = \Delta \underline{v}'_{Scul_{1m}} - \Delta \underline{v}'_{Scul_{2m}} \qquad \Delta \underline{v}'_{SculR_m} = \Delta \underline{v}'_{Scul_{1m}} + \Delta \underline{v}'_{Scul_{2m}} \qquad (8.2.2.1\text{-}34)$$

$$\Delta \underline{v}'_{SculQ_m} = \Delta \underline{v}'_{Scul_{3m}}$$

$$\Delta \underline{v}_{Scul_m} = \Delta \underline{v}'_{Scul_m} - LK_1 \, \Delta \underline{v}'_{Scul_m} - LK_2 \, \Delta \underline{v}'_{SculR_m} - LK_3 \, \Delta \underline{v}'_{SculQ_m}$$

where

$d\underline{\alpha}_{Cnt} = \underline{\omega}_{Puls} \, dt$ = Uncompensated angular rate sensor triad output differential pulse count vector.

$d\underline{\upsilon}_{Cnt} = \underline{a}_{SF_{Cnt}} \, dt$ = Uncompensated accelerometer triad output differential pulse count vector.

The algorithmic implementation of the Equations (8.2.2.1-33) $\Delta \underline{v}_{SculCnt_{1m}}$, $\Delta \underline{v}_{SculCnt_{2m}}$, $\Delta \underline{v}_{SculCnt_{3m}}$ integrations would be performed using the Equation (7.2.2.2.2-13) - (7.2.2.2.2-15) integration routines for $\Delta \underline{v}_{Scul_m}$ but with uncompensated data:

$$\underline{\alpha}_{Cnt_l}, \Delta \underline{\alpha}_{Cnt_l} = \text{From Equations (8.2.1.1-15)} \qquad (8.2.2.1\text{-}35)$$

$$\Delta \underline{\upsilon}_{Cnt_l} = \int_{t_{l-1}}^{t_l} d\underline{\upsilon}_{Cnt} \qquad \text{Summation Increment Of Accelerometer Output Pulses}$$

$$\underline{\upsilon}_{Cnt_l} = \underline{\upsilon}_{Cnt_{l-1}} + \Delta \underline{\upsilon}_{Cnt_l} \qquad (8.2.2.1\text{-}36)$$

$$\underline{\upsilon}_{Cnt_m} = \underline{\upsilon}_{Cnt_l}(t_l = t_m) \qquad \underline{\upsilon}_{Cnt_l} = 0 \quad \text{At } t = t_{m-1}.$$

$$\delta \underline{v}_{SculCnt_{1l}} = \frac{1}{2} \begin{bmatrix} \left(\alpha_{CntY_{l-1}} + \frac{1}{6} \Delta\alpha_{CntY_{l-1}}\right) \Delta\upsilon_{CntZ_l} - \left(\upsilon_{CntZ_{l-1}} + \frac{1}{6} \Delta\upsilon_{CntZ_{l-1}}\right) \Delta\alpha_{CntY_l} \\ \left(\alpha_{CntZ_{l-1}} + \frac{1}{6} \Delta\alpha_{CntZ_{l-1}}\right) \Delta\upsilon_{CntX_l} - \left(\upsilon_{CntX_{l-1}} + \frac{1}{6} \Delta\upsilon_{CntX_{l-1}}\right) \Delta\alpha_{CntZ_l} \\ \left(\alpha_{CntX_{l-1}} + \frac{1}{6} \Delta\alpha_{CntX_{l-1}}\right) \Delta\upsilon_{CntY_l} - \left(\upsilon_{CntY_{l-1}} + \frac{1}{6} \Delta\upsilon_{CntY_{l-1}}\right) \Delta\alpha_{CntX_l} \end{bmatrix} \qquad (8.2.2.1\text{-}37)$$

(Continued)

$$
\delta \underline{v}_{SculCnt2_l} = \frac{1}{2} \begin{bmatrix} \left(\alpha_{CntZ_{l-1}} + \frac{1}{6}\Delta\alpha_{CntZ_{l-1}}\right)\Delta\upsilon_{CntY_l} - \left(\upsilon_{CntY_{l-1}} + \frac{1}{6}\Delta\upsilon_{CntY_{l-1}}\right)\Delta\alpha_{CntZ_l} \\ \left(\alpha_{CntX_{l-1}} + \frac{1}{6}\Delta\alpha_{CntX_{l-1}}\right)\Delta\upsilon_{CntZ_l} - \left(\upsilon_{CntZ_{l-1}} + \frac{1}{6}\Delta\upsilon_{CntZ_{l-1}}\right)\Delta\alpha_{CntX_l} \\ \left(\alpha_{CntY_{l-1}} + \frac{1}{6}\Delta\alpha_{CntY_{l-1}}\right)\Delta\upsilon_{CntX_l} - \left(\upsilon_{CntX_{l-1}} + \frac{1}{6}\Delta\upsilon_{CntX_{l-1}}\right)\Delta\alpha_{CntY_l} \end{bmatrix}
$$

$$
\delta \underline{v}_{SculCnt3_l} = \frac{1}{2} \begin{bmatrix} \left(\alpha_{CntX_{l-1}} + \frac{1}{6}\Delta\alpha_{CntX_{l-1}}\right)\Delta\upsilon_{CntX_l} - \left(\upsilon_{CntX_{l-1}} + \frac{1}{6}\Delta\upsilon_{CntX_{l-1}}\right)\Delta\alpha_{CntX_l} \\ \left(\alpha_{CntY_{l-1}} + \frac{1}{6}\Delta\alpha_{CntY_{l-1}}\right)\Delta\upsilon_{CntY_l} - \left(\upsilon_{CntY_{l-1}} + \frac{1}{6}\Delta\upsilon_{CntY_{l-1}}\right)\Delta\alpha_{CntY_l} \\ \left(\alpha_{CntZ_{l-1}} + \frac{1}{6}\Delta\alpha_{CntZ_{l-1}}\right)\Delta\upsilon_{CntZ_l} - \left(\upsilon_{CntZ_{l-1}} + \frac{1}{6}\Delta\upsilon_{CntZ_{l-1}}\right)\Delta\alpha_{CntZ_l} \end{bmatrix}
\qquad \begin{matrix}(8.2.2.1\text{-}37)\\ (\text{Continued})\end{matrix}
$$

$$\Delta \underline{v}_{SculCnt1_l} = \Delta \underline{v}_{SculCnt1_{l-1}} + \delta \underline{v}_{SculCnt1_l}$$

$$\Delta \underline{v}_{SculCnt2_l} = \Delta \underline{v}_{SculCnt2_{l-1}} + \delta \underline{v}_{SculCnt2_l}$$

$$\Delta \underline{v}_{SculCnt3_l} = \Delta \underline{v}_{SculCnt3_{l-1}} + \delta \underline{v}_{SculCnt3_l}$$

$$\Delta \underline{v}_{SculCnt1_m} = \Delta \underline{v}_{SculCnt1_l}(t_l = t_m) \qquad \Delta \underline{v}_{SculCnt1_l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta \underline{v}_{SculCnt2_m} = \Delta \underline{v}_{SculCnt2_l}(t_l = t_m) \qquad \Delta \underline{v}_{SculCnt2_l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta \underline{v}_{SculCnt3_m} = \Delta \underline{v}_{SculCnt3_l}(t_l = t_m) \qquad \Delta \underline{v}_{SculCnt3_l} = 0 \quad \text{At } t = t_{m-1}$$

where

$\Delta\underline{\alpha}_{Cnt_l}$ = As defined in (8.2.1.1-15).

$\Delta\alpha_{CntX_l}, \Delta\alpha_{CntY_l}, \Delta\alpha_{CntZ_l}$ = X, Y, Z components of $\Delta\underline{\alpha}_{Cnt_l}$.

$\alpha_{CntX_l}, \alpha_{CntY_l}, \alpha_{CntZ_l}$ = X, Y, Z components of $\underline{\alpha}_{Cnt_l}$.

$\Delta\underline{\upsilon}_{Cnt_l}$ = As defined in (8.2.2.1-36).

$\Delta\upsilon_{CntX_l}, \Delta\upsilon_{CntY_l}, \Delta\upsilon_{CntZ_l}$ = X, Y, Z components of $\Delta\underline{\upsilon}_{Cnt_l}$.

$\upsilon_{CntX_l}, \upsilon_{CntY_l}, \upsilon_{CntZ_l}$ = X, Y, Z components of $\underline{\upsilon}_{Cnt_l}$.

It should be noted that in some applications, sufficient accuracy may be obtainable by recognizing that LK_1, LK_2, LK_3 as computed in (8.2.2.1-31) and (8.2.2.1-32) are small and potentially negligible in calculating $\Delta\underline{v}_{Scul_m}$. In this case the approximation can be made that

$\Delta\underline{v}_{Scul_m}$ equals $\Delta\underline{v}'_{Scul_m}$ in Equations (8.2.2.1-34) which also eliminates the requirement to calculate $\Delta\underline{v}'_{SculR_m}$, $\Delta\underline{v}'_{SculQ_m}$, $\Delta\underline{v}'_{Scul3_m}$ and $\Delta\underline{v}_{SculCnt3_m}$.

Equations (8.2.2.1-34) - (8.2.2.1-37) for $\Delta\underline{v}'_{Scul_{1m}}$, $\Delta\underline{v}'_{Scul_{2m}}$, $\Delta\underline{v}'_{Scul_{3m}}$ are based on the assumption that A_{Wt} and Ω_{Wt} can be approximated as constant over an m cycle. Under this assumption, if A_{Wt} or Ω_{Wt} contain non-linear terms (as addressed in Section 8.1.1.3), A_{Wt} and Ω_{Wt} would be calculated as a function of $\underline{\upsilon}_{Cnt_m}$ and $\underline{\alpha}_{Cnt_m}$. For the more general case in which A_{Wt} or Ω_{Wt} may have rapid variations, $\Delta\underline{v}'_{Scul_{1m}}$, $\Delta\underline{v}'_{Scul_{2m}}$, $\Delta\underline{v}'_{Scul_{3m}}$ can be calculated based on Equation (8.2.2-1), but computed from scale factor compensated inertial sensor output data $\underline{a}'_{SF}$, $\underline{\omega}'$ as in (8.2.2.1-1) and (8.2.2.1-3). Then the equivalent to (8.2.2.1-34) - (8.2.2.1-37) for $\Delta\underline{v}_{Scul_m}$ becomes:

$$\underline{\alpha}'_l,\ \Delta\underline{\alpha}'_l = \text{From Equations (8.2.1.1-18)} \tag{8.2.2.1-38}$$

$$\Delta\underline{\upsilon}_{Cnt_l} = \int_{t_{l-1}}^{t_l} d\underline{\upsilon}_{Cnt} \qquad \text{Summation Increment Of Accelerometer Output Pulses}$$

$$\Delta\underline{\upsilon}'_l = A_{Wt}\,\Delta\underline{\upsilon}_{Cnt_l} \tag{8.2.2.1-39}$$

$$\underline{\upsilon}'_l = \underline{\upsilon}'_{l-1} + \Delta\underline{\upsilon}'_l$$

$$\underline{\upsilon}'_m = \underline{\upsilon}'_l(t_l = t_m) \qquad \underline{\upsilon}'_l = 0 \quad \text{At } t = t_{m-1}.$$

$\delta\underline{v}'_{Scul_{1l}}$, $\delta\underline{v}'_{Scul_{2l}}$, $\delta\underline{v}'_{Scul_{3l}}$ = Equation (8.2.2.1-15) vector product forms of:

$$\delta\underline{v}'_{Scul_l} = \frac{1}{2}\left[\left(\underline{\alpha}'_{l-1} + \frac{1}{6}\,\Delta\underline{\alpha}'_{l-1}\right)\times\Delta\underline{\upsilon}'_l - \Delta\underline{\alpha}'_l\times\left(\underline{\upsilon}'_{l-1} + \frac{1}{6}\,\Delta\underline{\upsilon}'_{l-1}\right)\right]$$

$$\Delta\underline{v}'_{Scul_{1l}} = \Delta\underline{v}'_{Scul_{1l-1}} + \delta\underline{v}'_{Scul_{1l}}$$

$$\Delta\underline{v}'_{Scul_{2l}} = \Delta\underline{v}'_{Scul_{2l-1}} + \delta\underline{v}'_{Scul_{2l}} \tag{8.2.2.1-40}$$

$$\Delta\underline{v}'_{Scul_{3l}} = \Delta\underline{v}'_{Scul_{3l-1}} + \delta\underline{v}'_{Scul_{3l}}$$

(Continued)

$$\Delta \underline{v}'_{Scul_{1m}} = \Delta \underline{v}'_{Scul_{1l}}(t_l = t_m) \qquad \Delta \underline{v}'_{Scul_{1l}} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta \underline{v}'_{Scul_{2m}} = \Delta \underline{v}'_{Scul_{2l}}(t_l = t_m) \qquad \Delta \underline{v}'_{Scul_{2l}} = 0 \quad \text{At } t = t_{m-1} \qquad \text{(8.2.2.1-40) (Continued)}$$

$$\Delta \underline{v}'_{Scul_{3m}} = \Delta \underline{v}'_{Scul_{3l}}(t_l = t_m) \qquad \Delta \underline{v}'_{Scul_{3l}} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta \underline{v}'_{Scul_m} = \Delta \underline{v}'_{Scul_{1m}} - \Delta \underline{v}'_{Scul_{2m}} \qquad \Delta \underline{v}'_{SculR_m} = \Delta \underline{v}'_{Scul_{1m}} + \Delta \underline{v}'_{Scul_{2m}}$$

$$\Delta \underline{v}'_{SculQ_m} = \Delta \underline{v}'_{Scul_{3m}} \qquad \text{(8.2.2.1-41)}$$

$$\Delta \underline{v}_{Scul_m} = \Delta \underline{v}'_{Scul_m} - LK_1\, \Delta \underline{v}'_{Scul_m} - LK_2\, \Delta \underline{v}'_{SculR_m} - LK_3\, \Delta \underline{v}'_{SculQ_m}$$

Finally, if scale factor asymmetry compensation is being applied in the software algorithms as described in Section 8.1.1.3, Equations (8.2.2.1-38) - (8.2.2.1-39) would be based on $\underline{a}'_{SF}$ and $\underline{\omega}'$ given by Equations (8.1.1.3-19) and (8.1.1.3-21) rather than (8.2.2.1-3) and (8.2.2.1-1):

$$\underline{\omega}' = \Omega_{Wt_+}\, \underline{\omega}_{+Puls} + \Omega_{Wt_-}\, \underline{\omega}_{-Puls}$$

$$\underline{a}'_{SF} = A_{Wt_+}\, \underline{a}_{SF_{+Puls}} + A_{Wt_-}\, \underline{a}_{SF_{-Puls}} \qquad \text{(8.2.2.1-42)}$$

The A_{Wt_+}, A_{Wt_-}, Ω_{Wt_+}, Ω_{Wt_-} terms in Equation (8.2.2.1-42) are scale factor weighting matrices defined in Equations (8.1.1.3-20) and (8.1.1.3-22) for positive and negative uncompensated sensor pulse rate output data (See Section 8.1.1.3 for further clarification). Based on Equations (8.2.2.1-42), the equivalent to the (8.2.2.1-38) - (8.2.2.1-39) operations would be as follows:

$$\underline{\alpha}'_l,\ \Delta \underline{\alpha}'_l = \text{From Equations (8.2.1.1-22)} \qquad \text{(8.2.2.1-43)}$$

$$\Delta \underline{\upsilon}_{+Cnt_l} \equiv \int_{t_{l-1}}^{t_l} d\underline{\upsilon}_{+Cnt} \qquad \text{Summation Increment Of Positive Accelerometer Output Pulses}$$

$$\Delta \underline{\upsilon}_{-Cnt_l} \equiv \int_{t_{l-1}}^{t_l} d\underline{\upsilon}_{-Cnt} \qquad \text{Summation Increment Of Negative Accelerometer Output Pulses}$$

$$\Delta \underline{\upsilon}'_l = A_{Wt_+}\, \Delta \underline{\upsilon}_{+Cnt_l} + A_{Wt_-}\, \Delta \underline{\upsilon}_{-Cnt_l} \qquad \text{(8.2.2.1-44)}$$

$$\underline{\upsilon}'_l = \underline{\upsilon}'_{l-1} + \Delta \underline{\upsilon}'_l$$

$$\underline{\upsilon}'_m = \underline{\upsilon}'_l(t_l = t_m) \qquad \underline{\upsilon}'_l = 0 \quad \text{At } t = t_{m-1}$$

Equations (8.2.2.1-43) - (8.2.2.1-44) are based on the assumption that separate plus and minus pulse sum outputs are provided from the inertial sensor interface to the navigation software. If only a composite pulse sum is available (as in (8.2.2.1-38) - (8.2.2.1-39)), Equations (8.2.2.1-43) - (8.2.2.1-44) would be of the form:

$\underline{\alpha}'_l, \Delta\underline{\alpha}'_l$ = From Equations (8.2.1.1-23) (8.2.2.1-45)

$$\Delta\underline{\upsilon}_{Cnt_l} = \int_{t_{l-1}}^{t_l} d\underline{\upsilon}_{Cnt} \qquad \text{Summation Increment Of Accelerometer Output Pulses}$$

Do For Each Component Of $\Delta\underline{\upsilon}_{Cnt_l}$:

$$\text{If } \Delta\upsilon_{iCnt_l} \geq 0 \quad \text{Then:} \quad \Delta\upsilon'_{il} = A_{Wt_{i+}} \Delta\upsilon_{iCnt_l}$$
$$\text{Else:} \quad \Delta\upsilon'_{il} = A_{Wt_{i-}} \Delta\upsilon_{iCnt_l} \qquad (8.2.2.1\text{-}46)$$

$$\underline{\upsilon}'_l = \underline{\upsilon}'_{l-1} + \Delta\underline{\upsilon}'_l$$

$$\underline{\upsilon}'_m = \underline{\upsilon}'_l(t_l = t_m) \qquad \underline{\upsilon}'_l = 0 \quad \text{At } t = t_{m-1}$$

where

$\Delta\upsilon_{iCnt_l}$, $\Delta\upsilon'_{il}$ = Components i of $\underline{\upsilon}_{Cnt_l}$, $\Delta\underline{\upsilon}'_l$.

$A_{Wt_{i+}}$, $A_{Wt_{i-}}$ = Elements i, i in $A_{Wt\,+}$, $A_{Wt\,-}$.

8.2.2.2 COMBINED VELOCITY ROTATION COMPENSATION AND SCULLING INCREMENT COMPENSATION ALGORITHM

Following the same procedure leading to Equations (8.2.2.1-33) and (8.2.2.1-34) the following can be derived for the composite rotation-compensation/sculling term $\underline{v}_{Rot/Scul_m}$ defined analytically in Equations (8.2.2-2):

$$d\Delta\underline{v}_{Rot/SculCnt_1} = \begin{bmatrix} \alpha_{Cnt_Y}(t)\ d\upsilon_{Cnt_Z} \\ \alpha_{Cnt_Z}(t)\ d\upsilon_{Cnt_X} \\ \alpha_{Cnt_X}(t)\ d\upsilon_{Cnt_Y} \end{bmatrix} \qquad (8.2.2.2\text{-}1)$$

(Continued)

$$d\Delta\underline{v}_{Rot/SculCnt_2} = \begin{bmatrix} \alpha_{Cnt_Z}(t)\ \ d\upsilon_{Cnt_Y} \\ \alpha_{Cnt_X}(t)\ \ d\upsilon_{Cnt_Z} \\ \alpha_{Cnt_Y}(t)\ \ d\upsilon_{Cnt_X} \end{bmatrix}$$

$$d\Delta\underline{v}_{Rot/SculCnt_3} = \begin{bmatrix} \alpha_{Cnt_X}(t)\ d\ \upsilon_{Cnt_X} \\ \alpha_{Cnt_Y}(t)\ d\ \upsilon_{Cnt_Y} \\ \alpha_{Cnt_Z}(t)\ d\ \upsilon_{Cnt_Z} \end{bmatrix} \qquad \begin{matrix}(8.2.2.2\text{-}1)\\ \text{(Continued)}\end{matrix}$$

$$\Delta\underline{v}_{Rot/SculCnt_{1m}} = \int_{t_{m-1}}^{t_m} d\Delta\underline{v}_{Rot/SculCnt_1}$$

$$\Delta\underline{v}_{Rot/SculCnt_{2m}} = \int_{t_{m-1}}^{t_m} d\Delta\underline{v}_{Rot/SculCnt_2}$$

$$\Delta\underline{v}_{Rot/SculCnt_{3m}} = \int_{t_{m-1}}^{t_m} d\Delta\underline{v}_{Rot/SculCnt_3}$$

$$\Delta\underline{v}'_{Rot/Scul_{1m}} = \Omega A_{Wt_1}\ \Delta\underline{v}_{Rot/SculCnt_{1m}}$$

$$\Delta\underline{v}'_{Rot/Scul_{2m}} = \Omega A_{Wt_2}\ \Delta\underline{v}_{Rot/SculCnt_{2m}}$$

$$\Delta\underline{v}'_{Rot/Scul_{3m}} = \Omega A_{Wt_3}\ \Delta\underline{v}_{Rot/SculCnt_{3m}}$$

$$\Delta\underline{v}'_{Rot/Scul_m} = \Delta\underline{v}'_{Rot/Scul_{1m}} - \Delta\underline{v}'_{Rot/Scul_{2m}} \qquad (8.2.2.2\text{-}2)$$

$$\Delta\underline{v}'_{Rot/SculR_m} = \Delta\underline{v}'_{Rot/Scul_{1m}} + \Delta\underline{v}'_{Rot/Scul_{2m}}$$

$$\Delta\underline{v}'_{Rot/SculQ_m} = \Delta\underline{v}'_{Rot/Scul_{3m}}$$

$$\Delta\underline{v}_{Rot/Scul_m} = \Delta\underline{v}'_{Rot/Scul_m} - LK_1\ \Delta\underline{v}'_{Rot/Scul_m} - LK_2\ \Delta\underline{v}'_{Rot/SculR_m} - LK_3\ \Delta\underline{v}'_{Rot/SculQ_m}$$

It should be noted that in the development of Equations (8.2.2.2-2), angular rate sensor and accelerometer bias terms are dropped based on their smallness compared to the dominant scale factor and misalignment terms. In this respect, (8.2.2.2-2) are less than accurate than Equations (8.2.2.1-34) in which sensor bias terms cancel (see discussion following Equation (8.2.2.1-13)).

The algorithmic implementation of the Equations (8.2.2.2-1) $\Delta\underline{v}_{Rot/SculCnt_{1m}}$, $\Delta\underline{v}_{Rot/SculCnt_{2m}}$, $\Delta\underline{v}_{Rot/SculCnt_{3m}}$ integrations would be performed using the Equation (7.2.2.2.2-22) - (7.2.2.2.2-24) integration routines for $\Delta\underline{v}_{Rot/Scul_m}$ but with uncompensated data:

$$\underline{\alpha}_{Cnt_l},\ \Delta\underline{\alpha}_{Cnt_l} = \text{ From Equations (8.2.1.1-15)} \tag{8.2.2.2-3}$$

$$\underline{\upsilon}_{Cnt_l},\ \Delta\underline{\upsilon}_{Cnt_l} = \text{ As In Equations (8.2.2.1-36)} \tag{8.2.2.2-4}$$

$$\delta\underline{v}_{Rot/SculCnt_{1l}} = \begin{bmatrix} \left(\alpha_{CntY_{l-1}} + \frac{1}{2}\Delta\alpha_{CntY_l}\right)\Delta\upsilon_{CntZ_l} + \frac{1}{12}\left(\Delta\alpha_{CntY_{l-1}}\Delta\upsilon_{CntZ_l} - \Delta\upsilon_{CntZ_{l-1}}\Delta\alpha_{CntY_l}\right) \\ \left(\alpha_{CntZ_{l-1}} + \frac{1}{2}\Delta\alpha_{CntZ_l}\right)\Delta\upsilon_{CntX_l} + \frac{1}{12}\left(\Delta\alpha_{CntZ_{l-1}}\Delta\upsilon_{CntX_l} - \Delta\upsilon_{CntX_{l-1}}\Delta\alpha_{CntZ_l}\right) \\ \left(\alpha_{CntX_{l-1}} + \frac{1}{2}\Delta\alpha_{CntX_l}\right)\Delta\upsilon_{CntY_l} + \frac{1}{12}\left(\Delta\alpha_{CntX_{l-1}}\Delta\upsilon_{CntY_l} - \Delta\upsilon_{CntY_{l-1}}\Delta\alpha_{CntX_l}\right) \end{bmatrix}$$

$$\delta\underline{v}_{Rot/SculCnt_{2l}} = \begin{bmatrix} \left(\alpha_{CntZ_{l-1}} + \frac{1}{2}\Delta\alpha_{CntZ_l}\right)\Delta\upsilon_{CntY_l} + \frac{1}{12}\left(\Delta\alpha_{CntZ_{l-1}}\Delta\upsilon_{CntY_l} - \Delta\upsilon_{CntY_{l-1}}\Delta\alpha_{CntZ_l}\right) \\ \left(\alpha_{CntX_{l-1}} + \frac{1}{2}\Delta\alpha_{CntX_l}\right)\Delta\upsilon_{CntZ_l} + \frac{1}{12}\left(\Delta\alpha_{CntX_{l-1}}\Delta\upsilon_{CntZ_l} - \Delta\upsilon_{CntZ_{l-1}}\Delta\alpha_{CntX_l}\right) \\ \left(\alpha_{CntY_{l-1}} + \frac{1}{2}\Delta\alpha_{CntY_l}\right)\Delta\upsilon_{CntX_l} + \frac{1}{12}\left(\Delta\alpha_{CntY_{l-1}}\Delta\upsilon_{CntX_l} - \Delta\upsilon_{CntX_{l-1}}\Delta\alpha_{CntY_l}\right) \end{bmatrix} \tag{8.2.2.2-5}$$

$$\delta\underline{v}_{Rot/SculCnt_{3l}} = \begin{bmatrix} \left(\alpha_{CntX_{l-1}} + \frac{1}{2}\Delta\alpha_{CntX_l}\right)\Delta\upsilon_{CntX_l} + \frac{1}{12}\left(\Delta\alpha_{CntX_{l-1}}\Delta\upsilon_{CntX_l} - \Delta\upsilon_{CntX_{l-1}}\Delta\alpha_{CntX_l}\right) \\ \left(\alpha_{CntY_{l-1}} + \frac{1}{2}\Delta\alpha_{CntY_l}\right)\Delta\upsilon_{CntsY_l} + \frac{1}{12}\left(\Delta\alpha_{CntY_{l-1}}\Delta\upsilon_{CntY_l} - \Delta\upsilon_{CntY_{l-1}}\Delta\alpha_{CntY_l}\right) \\ \left(\alpha_{CntZ_{l-1}} + \frac{1}{2}\Delta\alpha_{CntZ_l}\right)\Delta\upsilon_{CntsZ_l} + \frac{1}{12}\left(\Delta\alpha_{CntZ_{l-1}}\Delta\upsilon_{CntZ_l} - \Delta\upsilon_{CntZ_{l-1}}\Delta\alpha_{CntZ_l}\right) \end{bmatrix}$$

(Continued)

$$\Delta\underline{v}_{Rot/SculCnt_{1l}} = \Delta\underline{v}_{Rot/SculCnt_{1l-1}} + \delta\underline{v}_{Rot/SculCnt_{1l}}$$

$$\Delta\underline{v}_{Rot/SculCnt_{2l}} = \Delta\underline{v}_{Rot/SculCnt_{2l-1}} + \delta\underline{v}_{Rot/SculCnt_{2l}}$$

$$\Delta\underline{v}_{Rot/SculCnt_{3l}} = \Delta\underline{v}_{Rot/SculCnt_{3l-1}} + \delta\underline{v}_{Rot/SculCnt_{3l}} \qquad \text{(8.2.2.2-5) (Continued)}$$

$$\Delta\underline{v}_{Rot/SculCnt_{1m}} = \Delta\underline{v}_{Rot/SculCnt_{1l}}(t_l = t_m) \qquad \Delta\underline{v}_{Rot/SculCnt_{1l}} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta\underline{v}_{Rot/SculCnt_{2m}} = \Delta\underline{v}_{Rot/SculCnt_{2l}}(t_l = t_m) \qquad \Delta\underline{v}_{Rot/SculCnt_{2l}} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta\underline{v}_{Rot/SculCnt_{3m}} = \Delta\underline{v}_{Rot/SculCnt_{3l}}(t_l = t_m) \qquad \Delta\underline{v}_{Rot/SculCnt_{3l}} = 0 \quad \text{At } t = t_{m-1}$$

It should also be noted that in some applications, sufficient accuracy may be obtainable by recognizing that LK_1, LK_2, LK_3 as computed in (8.2.2.1-31) and (8.2.2.1-32) are small and potentially negligible in calculating $\Delta\underline{v}_{Rot/Scul_m}$. In this case the approximation can be made that $\Delta\underline{v}_{Rot/Scul_m}$ equals $\Delta\underline{v}'_{Rot/Scul_m}$ in Equations (8.2.2.2-2) which also eliminates the requirement to calculate $\Delta\underline{v}'_{Rot/SculR_m}$, $\Delta\underline{v}'_{Rot/SculQ_m}$, $\Delta\underline{v}'_{Rot/Scul_{3m}}$ and $\Delta\underline{v}_{Rot/SculCnt_{3m}}$.

Equations (8.2.2.2-2) - (8.2.2.2-5) for $\Delta\underline{v}'_{Rot/Scul_{1m}}$, $\Delta\underline{v}'_{Rot/Scul_{2m}}$, $\Delta\underline{v}'_{Rot/Scul_{3m}}$ are based on the assumption that A_{Wt} and Ω_{Wt} can be approximated as constant over an m cycle. Under this assumption, if A_{Wt} or Ω_{Wt} contain non-linear terms (as addressed in Section 8.1.1.3), A_{Wt} and Ω_{Wt} would be calculated as a function of $\underline{\upsilon}_{Cnt_m}$ and $\underline{\alpha}_{Cnt_m}$. For the more general case in which A_{Wt} or Ω_{Wt} may have rapid variations, $\Delta\underline{v}'_{Rot/Scul_{1m}}$, $\Delta\underline{v}'_{Rot/Scul_{2m}}$, $\Delta\underline{v}'_{Rot/Scul_{3m}}$ can be calculated based on Equation (8.2.2-2), but from scale factor compensated inertial sensor output data $\underline{a}'_{SF}$, $\underline{\omega}'$ as in (8.2.2.1-1) and (8.2.2.1-3). Then the equivalent to (8.2.2.2-2) - (8.2.2.2-5) for $\Delta\underline{v}_{Rot/Scul_m}$ becomes:

$$\underline{\alpha}'_l,\ \Delta\underline{\alpha}'_l = \text{From Equations (8.2.1.1-18)} \qquad \text{(8.2.2.2-6)}$$

$$\underline{\upsilon}'_l,\ \Delta\underline{\upsilon}'_l = \text{As In Equations (8.2.2.1-39)} \qquad \text{(8.2.2.2-7)}$$

$\delta\underline{v}'_{Rot/Scul1_l}$, $\delta\underline{v}'_{Rot/Scul2_l}$, $\delta\underline{v}'_{Rot/Scul3_l}$ = Equation (8.2.2.1-15) vector product forms of:

$$\delta\underline{v}'_{Rot/Scul_l} = \left(\underline{\alpha}'_{l-1} + \frac{1}{2}\,\Delta\underline{\alpha}'_l\right) \times \Delta\underline{\upsilon}'_l + \frac{1}{12}\left(\Delta\underline{\alpha}'_{l-1} \times \Delta\underline{\upsilon}'_l - \Delta\underline{\alpha}'_l \times \Delta\underline{\upsilon}'_{l-1}\right)$$

$$\Delta\underline{v}'_{Rot/Scul1_l} = \Delta\underline{v}'_{Rot/Scul1_{l-1}} + \delta\underline{v}'_{Rot/Scul1_l}$$

$$\Delta\underline{v}'_{Rot/Scul2_l} = \Delta\underline{v}'_{Rot/Scul2_{l-1}} + \delta\underline{v}'_{Rot/Scul2_l} \qquad (8.2.2.2\text{-}8)$$

$$\Delta\underline{v}'_{Rot/Scul3_l} = \Delta\underline{v}'_{Rot/Scul3_{l-1}} + \delta\underline{v}'_{Rot/Scul3_l}$$

$$\Delta\underline{v}'_{Rot/Scul1_m} = \Delta\underline{v}'_{Rot/Scul1_l}(t_l = t_m) \qquad \Delta\underline{v}'_{Rot/Scul1_l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta\underline{v}'_{Rot/Scul2_m} = \Delta\underline{v}'_{Rot/Scul2_l}(t_l = t_m) \qquad \Delta\underline{v}'_{Rot/Scul2_l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta\underline{v}'_{Rot/Scul3_m} = \Delta\underline{v}'_{Rot/Scul3_l}(t_l = t_m) \qquad \Delta\underline{v}'_{Rot/Scul3_l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta\underline{v}'_{Rot/Scul_m} = \Delta\underline{v}'_{Rot/Scul1_m} - \Delta\underline{v}'_{Rot/Scul2_m}$$

$$\Delta\underline{v}'_{Rot/SculR_m} = \Delta\underline{v}'_{Rot/Scul1_m} + \Delta\underline{v}'_{Rot/Scul2_m}$$

$$\Delta\underline{v}'_{Rot/SculQ_m} = \Delta\underline{v}'_{Rot/Scul3_m} \qquad (8.2.2.2\text{-}9)$$

$$\Delta\underline{v}_{Scul_m} = \Delta\underline{v}'_{Rot/Scul_m} - LK_1\,\Delta\underline{v}'_{Rot/Scul_m} - LK_2\,\Delta\underline{v}'_{Rot/SculR_m} - LK_3\,\Delta\underline{v}'_{Rot/SculQ_m}$$

Finally, if scale factor asymmetry compensation is being applied in the software algorithms as described in Section 8.1.1.3, Equations (8.2.2.2-6) - (8.2.2.2-7) would be based on:

$$\underline{\alpha}'_l,\ \Delta\underline{\alpha}'_l = \text{From Equations (8.2.1.1-22)} \qquad (8.2.2.2\text{-}10)$$

$$\underline{\upsilon}'_l,\ \Delta\underline{\upsilon}'_l = \text{As In Equations (8.2.2.1-44)} \qquad (8.2.2.2\text{-}11)$$

Equations (8.2.2.2-10) - (8.2.2.2-11) are based on the assumption that separate plus and minus pulse sum outputs are provided from the inertial sensor interface to the navigation software. If only a composite pulse sum is available (as in (8.2.2.2-6) - (8.2.2.2-7)), Equations (8.2.2.2-10) - (8.2.2.2-11) would be of the form:

$$\underline{\alpha}'_l,\ \Delta\underline{\alpha}'_l = \text{From Equations (8.2.1.1-23)} \qquad (8.2.2.2\text{-}12)$$

$$\underline{\upsilon}'_l,\ \Delta\underline{\upsilon}'_l = \text{As In Equations (8.2.2.1-46)} \qquad (8.2.2.2\text{-}13)$$

8.2.3 INERTIAL SENSOR COMPENSATION FOR POSITION UPDATING

The accelerometer and angular rate sensor outputs in a strapdown system are used through suitable algorithms to update the system position data (Section 7.3). In general, this can be achieved by processing Equations (8.1.4.1-16), (7.3.3-10) and (7.3.3-11) (with (7.3.3.1-16) as an option to (7.3.3-11)) (repeated below) over an acceleration transformation update cycle to calculate a doubly integrated specific force acceleration increment in body B Frame coordinates. The doubly integrated acceleration increment is then used to update the position data. The scrolling vector obtained in these calculations is similar to but not identical to the scrolling portion of the position translation vector (analogous to the rotation vector) of Section 19.1, Equations (19.1.8-3) for updating position in Equations (19.1.5-9) based on a new unified strapdown algorithm concept. The scrolling portion of the position translation vector is defined as the integral of $\dot{\underline{\zeta}}_{Algo/c}$ in (19.1.8-3) minus the integrated $\underline{a}_{SF}$ portion of the $\underline{\eta}_{Algo/c}$ input.

$$\Delta\underline{R}^{B}_{SF_m} = \underline{S}_{\upsilon_m} + \Delta\underline{R}_{Rot_m} + \Delta\underline{R}_{Scrl_m} - \frac{1}{2}\,\delta\underline{v}_{Scul\text{-}SizeC_m}\,T_m$$

$$\Delta\underline{R}_{Rot_m} = \frac{1}{6}\left(\underline{S}_{\alpha_m}\times\underline{\upsilon}_m + \underline{\alpha}_m\times\underline{S}_{\upsilon_m}\right) \qquad \text{or Equation (7.3.3.1-16) as option}$$

$$\Delta\underline{R}_{Scrl_m} = \int_{t_{m-1}}^{t_m}\frac{1}{6}\left[6\,\Delta\underline{v}_{Scul}(t) - \underline{S}_{\alpha}(t)\times\underline{a}_{SF} + \underline{S}_{\upsilon}(t)\times\underline{\omega} + \underline{\alpha}(t)\times\underline{\upsilon}(t)\right]dt$$

$$\Delta\underline{v}_{Scul}(t) = \int_{t_{m-1}}^{t}\frac{1}{2}\left(\underline{\alpha}(\tau)\times\underline{a}_{SF} + \underline{\upsilon}(\tau)\times\underline{\omega}\right)d\tau \qquad (8.2.3\text{-}1)$$

$$\underline{S}_{\alpha}(t) = \int_{t_{m-1}}^{t}\underline{\alpha}(\tau)\,d\tau \qquad \underline{S}_{\alpha_m} = \underline{S}_{\alpha}(t_m)$$

$$\underline{S}_{\upsilon}(t) = \int_{t_{m-1}}^{t}\underline{\upsilon}(\tau)\,d\tau \qquad \underline{S}_{\upsilon_m} = \underline{S}_{\upsilon}(t_m)$$

$$\underline{\alpha}(\tau) = \int_{t_{m-1}}^{\tau}\underline{\omega}\,dt \qquad \underline{\alpha}_m = \underline{\alpha}(t_m)$$

$$\underline{\upsilon}(\tau) = \int_{t_{m-1}}^{\tau}\underline{a}_{SF}\,dt \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m)$$

where

$\Delta\underline{R}_{SF_m}$ = Doubly integrated specific force acceleration increment.

$\underline{a}_{SF}$ = Specific force acceleration vector sensed by accelerometer triad.

$\underline{\omega}$ = Angular rate vector sensed by angular rate sensor triad.

m = Acceleration-transformation/velocity-update/position-update cycle index.

$\underline{\upsilon}_m$ = Integrated specific force vector from t_{m-1} to t_m.

$\underline{\alpha}_m$ = Integrated angular rate vector from t_{m-1} to t_m.

$\underline{S}_{\upsilon_m}$ = Doubly integrated specific force vector from t_{m-1} to t_m.

$\underline{S}_{\alpha_m}$ = Doubly integrated angular rate vector from t_{m-1} to t_m.

$\Delta\underline{R}_{Rot_m}$ = "Position Rotation Compensation" term.

$\Delta\underline{v}_{Scul}(t)$ = "Sculling" term.

$\Delta\underline{R}_{Scrl_m}$ = "Scrolling" term.

All vectors in (8.2.3-1) are in B Frame coordinates, the superscript notation for which has been omitted for simplicity. Similarly, the IB subscript has been dropped from the $\underline{\omega}$ angular rate sensor vector which was identified in Chapter 7 as $\underline{\omega}_{IB}$.

Equations (8.2.3-1) are based on the use of compensated accelerometer and angular rate sensor data. Equations (8.1.2.1-3) (or (8.1.2.1-7)), (8.1.2.1-4), (8.1.2.1-5) (or (8.1.2.1-8)), (8.1.2.1-6), (8.1.2.2-3) (or (8.1.2.2-7)), (8.1.2.2-4), (8.1.2.2-5) (or (8.1.2.2-8)) and (8.1.2.2-6) provide the means for calculating the compensated form of $\underline{\alpha}_m$, $\underline{S}_{\alpha_m}$, $\underline{\upsilon}_m$ and $\underline{S}_{\upsilon_m}$ for Equations (8.2.3-1). The $\Delta\underline{R}_{Scrl_m}$ scrolling term in (8.2.3-1) can be calculated using compensated $\underline{\omega}$, $\underline{a}_{SF}$ angular rate, specific force acceleration input data based on compensation Equations (8.1.2.1-2) and (8.1.2.2-2) (similarly for integrating the scrolling portion of the position translation vector rate equation in (19.1.8-3) as part of position updating using the Chapter 19 (Section 19.1) unified approach - The scrolling portion of $\dot{\underline{\zeta}}_{Algo/c}$ is defined as $\dot{\underline{\zeta}}_{Algo/c}$ in (19.1.8-3) minus the integrated $\underline{a}_{SF}$ portion of the $\underline{\eta}_{Algo/c}$ input). A problem with this approach is that the $\underline{\omega}$, $\underline{a}_{SF}$ compensation operations would have to be performed at the high rate used to calculate $\Delta\underline{R}_{Scrl_m}$ (i.e., the *l* computer cycle rate as depicted in Equations (7.3.3.2-20) which is the computer implementation of Equations (8.2.3-1)). Alternatively, the Equations (7.3.3.2-20) scrolling algorithm can be calculated with uncompensated angular rate and accelerometer sensor data, with the uncompensated scrolling solution then compensated for sensor error at the position increment update rate (i.e., the m rate in Equations (7.3.3.2-20)). The following section describes the latter approach.

8.2.3.1 SCROLLING INCREMENT AND DOUBLE INTEGRATION TERM COMPENSATION ALGORITHMS

Following the identical procedure that led to Equations (8.2.2.1-33) - (8.2.2.1-34) for sculling compensation, we find for the Equation (8.2.3-1) scrolling term compensation:

$d\Delta\underline{R}_{ScrlCnt_1},\ d\Delta\underline{R}_{ScrlCnt_2},\ d\Delta\underline{R}_{ScrlCnt_3}$ = Equation (8.2.2.1-15) vector product forms of:

$$d\Delta\underline{R}_{ScrlCnt} = \frac{1}{6}\Big[6\,\Delta\underline{v}_{SculCnt}(t)\,dt - \underline{S}_{\alpha Cnt}(t)\times d\underline{\upsilon}_{Cnt} - d\underline{\alpha}_{Cnt}\times\underline{S}_{\upsilon Cnt}(t) + \underline{\alpha}_{Cnt}(t)\times\underline{\upsilon}_{Cnt}(t)\,dt\Big]$$

with

$$\Delta\underline{v}_{SculCnt}(t) = \int_{t_{m-1}}^{t}\frac{1}{2}\Big(\underline{\alpha}_{Cnt}(\tau)\times d\underline{\upsilon}_{Cnt} - d\underline{\alpha}_{Cnt}\times\underline{\upsilon}_{Cnt}(\tau)\Big) \qquad (8.2.3.1\text{-}1)$$

$$\Delta\underline{R}_{ScrlCnt_{1m}} = \int_{t_{m-1}}^{t_m} d\Delta\underline{R}_{ScrlCnt_1} \qquad \Delta\underline{R}_{ScrlCnt_{2m}} = \int_{t_{m-1}}^{t_m} d\Delta\underline{R}_{ScrlCnt_2}$$

$$\Delta\underline{R}_{ScrlCnt_{3m}} = \int_{t_{m-1}}^{t_m} d\Delta\underline{R}_{ScrlCnt_3}$$

$$\Delta\underline{R}'_{Scrl_{1m}} = \Omega A_{Wt_1}\,\Delta\underline{R}_{ScrlCnt_{1m}} \qquad \Delta\underline{R}'_{Scrl_{2m}} = \Omega A_{Wt_2}\,\Delta\underline{R}_{ScrlCnt_{2m}}$$

$$\Delta\underline{R}'_{Scrl_{3m}} = \Omega A_{Wt_3}\,\Delta\underline{R}_{ScrlCnt_{3m}}$$

$$\Delta\underline{R}'_{Scrl_m} = \Delta\underline{R}'_{Scrl_{1m}} - \Delta\underline{R}'_{Scrl_{2m}} \qquad \Delta\underline{R}'_{ScrlR_m} = \Delta\underline{R}'_{Scrl_{1m}} + \Delta\underline{R}'_{Scrl_{2m}} \qquad (8.2.3.1\text{-}2)$$

$$\Delta\underline{R}'_{ScrlQ_m} = \Delta\underline{R}'_{Scrl_{3m}}$$

$$\Delta\underline{R}_{Scrl_m} = \Delta\underline{R}'_{Scrl_m} - LK_1\,\Delta\underline{R}'_{Scrl_m} - LK_2\,\Delta\underline{R}'_{ScrlR_m} - LK_3\,\Delta\underline{R}'_{ScrlQ_m}$$

where

$d\underline{\alpha}_{Cnt},\ d\underline{\upsilon}_{Cnt} = \underline{\omega}\,dt,\ \underline{a}_{SF}\,dt$ = Uncompensated angular rate sensor and accelerometer triad output differential pulse count vectors.

$\underline{\alpha}_{Cnt},\ \underline{S}_{\alpha Cnt},\ \underline{\upsilon}_{Cnt},\ \underline{S}_{\upsilon Cnt},\ \Delta\underline{v}_{SculCnt}$ = $\underline{\alpha}$, $\underline{S}_{\alpha}$, $\underline{\upsilon}$, $\underline{S}_{\upsilon}$, $\Delta\underline{v}_{Scul}$ calculated from uncompensated inertial sensor data.

$\Delta \underline{R}_{ScrlCnt}$ = $\Delta \underline{R}_{Scrl}$ in Equations (8.2.3-1) calculated with uncompensated inertial sensor data.

$d\Delta \underline{R}_{ScrlCnt}$ = Differential change in $\Delta \underline{R}_{ScrlCnt}$.

It should be noted that in the development of Equations (8.2.3.1-2), angular rate sensor and accelerometer bias terms are dropped based on their smallness compared to the dominant scale factor and misalignment terms. In this respect, the development of (8.2.3.1-2) differs from the development of Equations (8.2.2.1-34) in which sensor bias terms cancel (see discussion following Equation (8.2.2.1-13)).

The algorithmic implementation of the Equations (8.2.3.1-1) $\Delta \underline{R}_{ScrlCnt_{1m}}$, $\Delta \underline{R}_{ScrlCnt_{2m}}$, $\Delta \underline{R}_{ScrlCnt_{3m}}$ integrations would be performed using the Equations (7.3.3.2-20) integration routine for $\Delta \underline{R}_{Scrl_m}$, but with uncompensated data. Including the uncompensated version of Equations (7.3.3.2-18) - (7.3.3.2-19) for the $\underline{S}_{\alpha_m}$, $\underline{S}_{\upsilon_m}$ algorithms finds:

$$\underline{\alpha}_{Cnt_l}, \Delta\underline{\alpha}_{Cnt_l}, \underline{\upsilon}_{Cnt_l}, \Delta\underline{\upsilon}_{Cnt_l} = \begin{array}{c}\text{From Equations}\\ \text{(8.2.1.1-15) and (8.2.2.1-36)}\end{array} \tag{8.2.3.1-3}$$

$$\Delta \underline{S}_{\alpha Cnt_l} = \underline{\alpha}_{Cnt_{l-1}} T_l + \frac{T_l}{12}\left(5\, \Delta\underline{\alpha}_{Cnt_l} + \Delta\underline{\alpha}_{Cnt_{l-1}}\right)$$

$$\Delta \underline{S}_{\upsilon Cnt_l} = \underline{\upsilon}_{Cnt_{l-1}} T_l + \frac{T_l}{12}\left(5\, \Delta\underline{\upsilon}_{Cnt_l} + \Delta\underline{\upsilon}_{Cnt_{l-1}}\right) \tag{8.2.3.1-4}$$

$$\underline{S}_{\alpha Cnt_l} = \underline{S}_{\alpha Cnt_{l-1}} + \Delta \underline{S}_{\alpha Cnt_l} \qquad \underline{S}_{\upsilon Cnt_l} = \underline{S}_{\upsilon Cnt_{l-1}} + \Delta \underline{S}_{\upsilon Cnt_l}$$

$$\underline{S}_{\alpha Cnt_m} = \underline{S}_{\alpha Cnt_l}(t_l = t_m) \qquad \underline{S}_{\upsilon Cnt_m} = \underline{S}_{\upsilon Cnt_l}(t_l = t_m)$$

$$\underline{S}_{\alpha Cnt_l} = 0 \quad \text{At } t = t_{m-1}. \qquad \underline{S}_{\upsilon Cnt_l} = 0 \quad \text{At } t = t_{m-1}$$

$\delta \underline{R}_{ScrlCnt_{1l}}, \delta \underline{R}_{ScrlCnt_{2l}}, \delta \underline{R}_{ScrlCnt_{3l}}$ = Equation (8.2.2.1-15) vector product forms of the following $\delta \underline{R}_{ScrlCnt_l}$ expression using Equations (8.2.2.1-37) for the $\Delta \underline{v}_{SculCnt_{1l}}$, $\Delta \underline{v}_{SculCnt_{2l}}$, $\Delta \underline{v}_{SculCnt_{3l}}$ sculling terms:

$$\delta \underline{R}_{ScrlCnt_l} = \delta \underline{R}_{ScrlCntA_l} + \delta \underline{R}_{ScrlCntB_l} \tag{8.2.3.1-5}$$

with

(Continued)

$$
\begin{aligned}
\delta \underline{R}_{ScrlCntA\,l} = {} & \Delta \underline{v}_{SculCnt_{l-1}}\, T_l \\
& + \frac{1}{2} \left[\underline{\alpha}_{Cnt_{l-1}} - \frac{1}{12} \left(\Delta \underline{\alpha}_{Cnt_l} - \Delta \underline{\alpha}_{Cnt_{l-1}} \right) \right] \times \left(\Delta \underline{S}_{\upsilon Cnt_l} - \underline{\upsilon}_{Cnt_{l-1}}\, T_l \right) \\
& - \frac{1}{2} \left(\Delta \underline{S}_{\alpha Cnt_l} - \underline{\alpha}_{Cnt_{l-1}}\, T_l \right) \times \left[\underline{\upsilon}_{Cnt_{l-1}} - \frac{1}{12} \left(\Delta \underline{\upsilon}_{Cnt_l} - \Delta \underline{\upsilon}_{Cnt_{l-1}} \right) \right]
\end{aligned}
$$

$$
\begin{aligned}
\delta \underline{R}_{ScrlCntB_l} = {} & - \frac{1}{6}\, \Delta \underline{\alpha}_{Cnt_l} \times \left[\underline{S}_{\upsilon Cnt_{l-1}} + \frac{T_l}{24} \left(\Delta \underline{\upsilon}_{Cnt_l} - \Delta \underline{\upsilon}_{Cnt_{l-1}} \right) \right] \\
& - \frac{1}{6} \left[\underline{S}_{\alpha Cnt_{l-1}} + \frac{T_l}{24} \left(\Delta \underline{\alpha}_{Cnt_l} - \Delta \underline{\alpha}_{Cnt_{l-1}} \right) \right] \times \Delta \underline{\upsilon}_{Cnt_l} \\
& + \frac{T_l}{6} \left[\underline{\alpha}_{Cnt_{l-1}} - \frac{1}{6} \left(\Delta \underline{\alpha}_{Cnt_l} - \Delta \underline{\alpha}_{Cnt_{l-1}} \right) \right] \times \left[\underline{\upsilon}_{Cnt_{l-1}} - \frac{1}{6} \left(\Delta \underline{\upsilon}_{Cnt_l} - \Delta \underline{\upsilon}_{Cnt_{l-1}} \right) \right] \\
& - \frac{T_l}{2160} \left(\Delta \underline{\alpha}_{Cnt_l} - \Delta \underline{\alpha}_{Cnt_{l-1}} \right) \times \left(\Delta \underline{\upsilon}_{Cnt_l} - \Delta \underline{\upsilon}_{Cnt_{l-1}} \right)
\end{aligned}
$$

(8.2.3.1-5)
(Continued)

$$\Delta \underline{R}_{ScrlCnt\,1l} = \Delta \underline{R}_{ScrlCnt\,1l-1} + \delta \underline{R}_{ScrlCnt\,1l}$$

$$\Delta \underline{R}_{ScrlCnt\,2l} = \Delta \underline{R}_{ScrlCnt\,2l-1} + \delta \underline{R}_{ScrlCnt\,2l}$$

$$\Delta \underline{R}_{ScrlCnt\,3l} = \Delta \underline{R}_{ScrlCnt\,3l-1} + \delta \underline{R}_{ScrlCnt\,3l}$$

$$\Delta \underline{R}_{ScrlCnt\,1m} = \Delta \underline{R}_{ScrlCnt\,1l}(t_l = t_m) \qquad \Delta \underline{R}_{ScrlCnt\,1l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta \underline{R}_{ScrlCnt\,2m} = \Delta \underline{R}_{ScrlCnt\,2l}(t_l = t_m) \qquad \Delta \underline{R}_{ScrlCnt\,2l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta \underline{R}_{ScrlCnt\,3m} = \Delta \underline{R}_{ScrlCnt\,3l}(t_l = t_m) \qquad \Delta \underline{R}_{ScrlCnt\,3l} = 0 \quad \text{At } t = t_{m-1}$$

Equations (8.2.3.1-2) - (8.2.3.1-5) for $\Delta \underline{R}'_{Scrl_{1m}}, \Delta \underline{R}'_{Scrl_{2m}}, \Delta \underline{R}'_{Scrl_{3m}}$ are based on the assumption that A_{Wt} and Ω_{Wt} can be approximated as constant over an m cycle. Under this assumption, if A_{Wt} or Ω_{Wt} contain non-linear terms (as addressed in Section 8.1.1.3), A_{Wt} and Ω_{Wt} would be calculated as a function of $\underline{\upsilon}_{Cnt_m}$ and $\underline{\alpha}_{Cnt_m}$. For the more general case in which A_{Wt} or Ω_{Wt} may have rapid variations, $\Delta \underline{R}'_{Scrl_{1m}}, \Delta \underline{R}'_{Scrl_{2m}}, \Delta \underline{R}'_{Scrl_{3m}}$ can be calculated based on Equation (8.2.3-1), but computed from scale factor compensated inertial sensor output data $\underline{a}'_{SF}$, $\underline{\omega}'$ as in (8.2.2.1-1) and (8.2.2.1-3). Then the equivalent to (8.2.3.1-2) - (8.2.3.1-5) for $\Delta \underline{R}_{Scrl_m}$ becomes:

$$\underline{\alpha}'_l,\ \Delta \underline{\alpha}'_l,\ \underline{\upsilon}'_l,\ \Delta \underline{\upsilon}'_l = \begin{array}{c}\text{From Equations} \\ \text{(8.2.1.1-18) and (8.2.2.1-39)}\end{array} \qquad (8.2.3.1\text{-}6)$$

$$\Delta \underline{S}'_{\alpha_l} = \underline{\alpha}'_{l-1}\, T_l + \frac{T_l}{12}\left(5\, \Delta\underline{\alpha}'_l + \Delta\underline{\alpha}'_{l-1}\right)$$

$$\Delta \underline{S}'_{\upsilon_l} = \underline{\upsilon}'_{l-1}\, T_l + \frac{T_l}{12}\left(5\, \Delta\underline{\upsilon}'_l + \Delta\underline{\upsilon}'_{l-1}\right)$$

$$\underline{S}'_{\alpha_l} = \underline{S}'_{\alpha_{l-1}} + \Delta\underline{S}'_{\alpha_l} \qquad \underline{S}'_{\upsilon_l} = \underline{S}'_{\upsilon_{l-1}} + \Delta\underline{S}'_{\upsilon_l} \tag{8.2.3.1-7}$$

$$\underline{S}'_{\alpha_m} = \underline{S}'_{\alpha_l}(t_l = t_m) \qquad \underline{S}'_{\upsilon_m} = \underline{S}'_{\upsilon_l}(t_l = t_m)$$

$$\underline{S}'_{\alpha_l} = 0 \quad \text{At } t = t_{m-1}. \qquad \underline{S}'_{\upsilon_l} = 0 \quad \text{At } t = t_{m-1}.$$

$\delta\underline{R}'_{Scrl1_l}, \delta\underline{R}'_{Scrl2_l}, \delta\underline{R}'_{Scrl3_l}$ = Equation (8.2.2.1-15) vector product forms of the following $\delta\underline{R}'_{Scrl_l}$ expression using Equations (8.2.2.1-40) for the $\Delta\underline{v}'_{Scul1_l}$, $\Delta\underline{v}'_{Scul2_l}$, $\Delta\underline{v}'_{Scul3_l}$ versions of the $\Delta\underline{v}'_{Scul_l}$ sculling term:

$$\delta\underline{R}'_{Scrl_l} = \delta\underline{R}'_{ScrlA_l} + \delta\underline{R}'_{ScrlB_l}$$

with

$$\begin{aligned}
\delta\underline{R}'_{ScrlA_l} = {} & \Delta\underline{v}'_{Scul_{l-1}}\, T_l + \frac{1}{2}\left[\underline{\alpha}'_{l-1} - \frac{1}{12}\left(\Delta\underline{\alpha}'_l - \Delta\underline{\alpha}'_{l-1}\right)\right] \times \left(\Delta\underline{S}'_{\upsilon_l} - \underline{\upsilon}'_{l-1}\, T_l\right) \\
& - \frac{1}{2}\left(\Delta\underline{S}'_{\alpha_l} - \underline{\alpha}'_{l-1}\, T_l\right) \times \left[\underline{\upsilon}'_{l-1} - \frac{1}{12}\left(\Delta\underline{\upsilon}'_l - \Delta\underline{\upsilon}'_{l-1}\right)\right] \\
\delta\underline{R}'_{ScrlB_l} = {} & -\frac{1}{6}\,\Delta\underline{\alpha}'_l \times \left[\underline{S}'_{\upsilon_{l-1}} + \frac{T_l}{24}\left(\Delta\underline{\upsilon}'_l - \Delta\underline{\upsilon}'_{l-1}\right)\right] \\
& - \frac{1}{6}\left[\underline{S}'_{\alpha_{l-1}} + \frac{T_l}{24}\left(\Delta\underline{\alpha}'_l - \Delta\underline{\alpha}'_{l-1}\right)\right] \times \Delta\underline{\upsilon}'_l \\
& + \frac{T_l}{6}\left[\underline{\alpha}'_{l-1} - \frac{1}{6}\left(\Delta\underline{\alpha}'_l - \Delta\underline{\alpha}'_{l-1}\right)\right] \times \left[\underline{\upsilon}'_{l-1} - \frac{1}{6}\left(\Delta\underline{\upsilon}'_l - \Delta\underline{\upsilon}'_{l-1}\right)\right] \\
& - \frac{T_l}{2160}\left(\Delta\underline{\alpha}'_l - \Delta\underline{\alpha}'_{l-1}\right) \times \left(\Delta\underline{\upsilon}'_l - \Delta\underline{\upsilon}'_{l-1}\right)
\end{aligned} \tag{8.2.3.1-8}$$

$$\Delta\underline{R}'_{Scrl1_l} = \Delta\underline{R}'_{Scrl1_{l-1}} + \delta\underline{R}'_{Scrl1_l} \qquad \Delta\underline{R}'_{Scrl2_l} = \Delta\underline{R}'_{Scrl2_{l-1}} + \delta\underline{R}'_{Scrl2_l}$$

$$\Delta\underline{R}'_{Scrl3_l} = \Delta\underline{R}'_{Scrl3_{l-1}} + \delta\underline{R}'_{Scrl3_l}$$

(Continued)

$$\Delta\underline{R}'_{Scrl1_m} = \Delta\underline{R}'_{Scrl1_l}(t_l = t_m) \qquad \Delta\underline{R}'_{Scrl1_l} = 0 \quad \text{At } t = t_{m-1}$$
$$\Delta\underline{R}'_{Scrl2_m} = \Delta\underline{R}'_{Scrl2_l}(t_l = t_m) \qquad \Delta\underline{R}'_{Scrl2_l} = 0 \quad \text{At } t = t_{m-1} \qquad \text{(8.2.3.1-8) (Continued)}$$
$$\Delta\underline{R}'_{Scrl3_m} = \Delta\underline{R}'_{Scrl3_l}(t_l = t_m) \qquad \Delta\underline{R}'_{Scrl3_l} = 0 \quad \text{At } t = t_{m-1}$$

$$\Delta\underline{R}'_{Scrl_m} = \Delta\underline{R}'_{Scrl1_m} - \Delta\underline{R}'_{Scrl2_m} \qquad \Delta\underline{R}'_{ScrlR_m} = \Delta\underline{R}'_{Scrl1_m} + \Delta\underline{R}'_{Scrl2_m}$$
$$\Delta\underline{R}'_{ScrlQ_m} = \Delta\underline{R}'_{Scrl3_m} \qquad \text{(8.2.3.1-9)}$$
$$\Delta\underline{R}_{Scrl_m} = \Delta\underline{R}'_{Scrl_m} - LK_1\, \Delta\underline{R}'_{Scrl_m} - LK_2\, \Delta\underline{R}'_{ScrlR_m} - LK_3\, \Delta\underline{R}'_{ScrlQ_m}$$

Finally, if scale factor asymmetry compensation is being applied in the software algorithms as described in Section 8.1.1.3, Equations (8.2.3.1-6) would be based on:

$$\underline{\alpha}'_l,\ \Delta\underline{\alpha}'_l,\ \underline{\upsilon}'_l,\ \Delta\underline{\upsilon}'_l = \begin{array}{c}\text{From Equations}\\ \text{(8.2.1.1-22) and (8.2.2.1-44)}\end{array} \qquad \text{(8.2.3.1-10)}$$

Equations (8.2.3.1-10) are based on the assumption that separate plus and minus pulse sum outputs are provided from the inertial sensor interface to the navigation software. If only a composite pulse sum is available (as in (8.2.3.1-3)), Equations (8.2.3.1-10) would be of the form:

$$\underline{\alpha}'_l,\ \Delta\underline{\alpha}'_l,\ \underline{\upsilon}'_l,\ \Delta\underline{\upsilon}'_l = \begin{array}{c}\text{From Equations}\\ \text{(8.2.1.1-23) and (8.2.2.1-46)}\end{array} \qquad \text{(8.2.3.1-11)}$$

8.3 SENSOR ASSEMBLY ALIGNMENT COMPENSATION

The attitude of the vehicle in which the strapdown inertial navigation system (INS) is installed is determined from the attitude direction matrix $\left(C_B^L\right)$, the inertial sensor assembly mounting misalignments and the orientation of the INS mount relative to user vehicle reference axes. The following coordinate frame definitions apply:

L = Locally level attitude reference frame.

B = Body (or inertial sensor assembly) coordinate frame.

M = INS mount coordinate frame (the B Frame is nominally aligned to the M Frame).

VRF = User vehicle reference axes.

An attitude direction cosine matrix relating the user vehicle and locally level attitude reference axes can be written from the Equation (3.2.1-5) chain law and (3.2.1-3):

$$C_{VRF}^{L} = C_{B}^{L}\left(C_{B}^{M}\right)^{T} C_{VRF}^{M} \tag{8.3-1}$$

Using Equation (3.2.2.1-8), the C_B^M direction cosine matrix in (8.3-1) can be defined in terms of the associated rotation vector components as follows:

$$C_{B}^{M} = I + \frac{\sin J}{J}\left(\underline{J} \times\right) + \frac{(1 - \cos J)}{J^2}\left(\underline{J} \times\right)\left(\underline{J} \times\right) \approx I + \left(\underline{J} \times\right) \tag{8.3-2}$$

where

$\underline{J}$, J = Sensor triad mount misalignment rotation error vector and its magnitude (determined using the Chapter 18, Section 18.4.7.4 system test procedure).

The C_{VRF}^M matrix in Equation (8.3-1) is a function of the particular mount orientation in the user vehicle.

Once C_{VRF}^L is obtained from Equation (8.3-1), the typical roll, pitch, heading outputs of the user vehicle can be easily calculated using Equations (4.1.2-1) and (4.1.2-2) with the C_{ij} terms interpreted as the elements of the C_{VRF}^L matrix.

In some applications, vector data defined in the B Frame or the locally level navigation N Frame is also desired in VRF coordinates (e.g., B Frame angular rate and specific force acceleration, and N Frame velocity). This is easily achieved using the following direction cosine transformation matrices:

$$C_{B}^{VRF} = \left(C_{VRF}^{M}\right)^{T} C_{B}^{M} \qquad C_{N}^{VRF} = \left[C_{L}^{N}\, C_{B}^{L}\left(C_{B}^{M}\right)^{T} C_{VRF}^{M}\right]^{T} \tag{8.3-3}$$

The C_L^N matrix in Equation (8.3-3) is provided by Equation (4.1.1-2) repeated below:

$$C_{L}^{N} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{8.3-4}$$

8.4 STRAPDOWN INERTIAL SENSOR ASSEMBLY COMPENSATION ALGORITHM SUMMARY

Table 8.4-1 illustrates how the Chapter 8 inertial sensor assembly compensation algorithms would be applied to the Table 7.5-1 navigation algorithms which were derived based on perfect inertial sensor assembly input data. Table 8.4-1 lists the algorithm function, input parameters, output parameters and equation number. Definitions for the input/output parameters are provided in the sections in which the algorithms/equations are derived as listed in the Parameter Index tabulation in the back of the book. The Chapter 8 algorithms selected for Table 8.4-1 provide for general inertial sensor data scale factor compensation at the high speed input rate, and for accelerometer scale factor asymmetry compensation as described in Section 8.1.1.3. Section 8.1.3.1 pulse count residual quantization compensation is shown for the accelerometers, while Section 8.1.3.2 turn-around dead-band quantization compensation is shown for the angular rate sensors. This quantization compensation approach has been used in some inertial sensor assemblies containing electrically rebalanced pendulous accelerometers and ring laser gyro type angular rate sensors (both described in Reference 31).

Table 8.4-1 Summary Of Typical Strapdown Inertial Navigation System Computations With Sensor Compensation Algorithms

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
HIGH SPEED CALCULATIONS			
Scale Factor Compensated Integrated B Frame Angular Rate Increments	Ω_{Wt}, $\Delta\underline{\alpha}_{Cnt_l}$	$\Delta\underline{\alpha}'_l$, $\underline{\alpha}'_l$, $\underline{\alpha}'_m$	(8.2.1.1-18)
Scale Factor Compensated Integrated B Frame Acceleration Increments	A_{Wt_+}, A_{Wt_-}, $\Delta\underline{\upsilon}_{+Cnt_l}$, $\Delta\underline{\upsilon}_{-Cnt_l}$	$\Delta\underline{\upsilon}'_l$, $\underline{\upsilon}'_l$, $\underline{\upsilon}'_m$	(8.2.2.1-44)
Scale Factor Compensated Coning Increment	$\Delta\underline{\alpha}'_l$, $\underline{\alpha}'_l$	$\underline{\beta}'_m$	(8.2.1.1-19)

(Continued)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Scale Factor Compensated Sculling Increments	$\Delta\underline{\alpha}'_l$, $\underline{\alpha}'_l$, $\Delta\underline{\upsilon}'_l$, $\underline{\upsilon}'_l$	$\Delta\underline{v}'_{Scul_{1l}}$, $\Delta\underline{v}'_{Scul_{1m}}$, $\Delta\underline{v}'_{Scul_{2l}}$, $\Delta\underline{v}'_{Scul_{2m}}$, $\Delta\underline{v}'_{Scul_{3l}}$, $\Delta\underline{v}'_{Scul_{3m}}$	(8.2.2.1-40)
High Speed Update Time Interval (Constant)	Constant	T_l	—
Scale Factor Compensated Doubly Integrated B Frame Angular Rate And Acceleration Increments (For High Resolution Position Algorithm)	$\Delta\underline{\alpha}'_l$, $\underline{\alpha}'_l$, $\Delta\underline{\upsilon}'_l$, $\underline{\upsilon}'_l$, T_l	$\Delta\underline{S}'_{\alpha_l}$, $\underline{S}'_{\alpha_l}$, $\underline{S}'_{\alpha_m}$, $\Delta\underline{S}'_{\upsilon_l}$, $\underline{S}'_{\upsilon_l}$, $\underline{S}'_{\upsilon_m}$	(8.2.3.1-7)
Scale Factor Compensated Scrolling Increment (For High Resolution Position Algorithm)	$\Delta\underline{\alpha}'_l$, $\underline{\alpha}'_l$, $\Delta\underline{S}'_{\alpha_l}$, $\underline{S}'_{\alpha_l}$, $\Delta\underline{\upsilon}'_l$, $\underline{\upsilon}'_l$, $\Delta\underline{S}'_{\upsilon_l}$, $\underline{S}'_{\upsilon_l}$, $\Delta\underline{v}'_{Scul_{1l}}$, $\Delta\underline{v}'_{Scul_{2l}}$, $\Delta\underline{v}'_{Scul_{3l}}$, T_l	$\Delta\underline{R}'_{Scrl_{1m}}$, $\Delta\underline{R}'_{Scrl_{2m}}$, $\Delta\underline{R}'_{Scrl_{3m}}$	(8.2.3.1-8)
Scale Factor Compensated Accelerometer Size Effect And Anisoinertia Input Parameters	$\Delta\underline{\alpha}'_l$	η_{ij_m}, $\Delta\underline{\alpha}'_m$, $\Delta\underline{\alpha}'_{(l\,:\,m)\,-1}$	(8.1.4.1.4-5)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
NORMAL SPEED CALCULATIONS			
INERTIAL SENSOR COMPENSATION			
Angular Rate Sensor Calibration Data	Sensor And System Level Test Results. May Include Active Temperature Functional Dependency.	$F_{SensScal}$, $\delta\underline{\omega}_{SensBias}$, $F_{SensAlgn}$, $\kappa_{SystScal/Mis}$, $\underline{\kappa}_{SystBias}$	—
Angular Rate Sensor Scale Factor And Misalignment Compensation Terms	$F_{SensScal}$, $F_{SensAlgn}$, $\kappa_{SystScal/Mis}$	F_{Scal}, F_{Algn}	(8.1.1.1.1-13), (8.1.1.1.1-14), (8.1.1.1.1-16)
Nominal Angular Rate Sensor Scale Factor (Constant)	Constant	Ω_{Wt_0}	—
Angular Rate Sensor Scale Factor Matrix And Misalignment Compensation Coefficients	Ω_{Wt_0}, F_{Scal}, F_{Algn}	Ω_{Wt}, K_{Mis}	(8.1.1.1-5), (8.1.1.1-6)
Angular Rate Sensor Bias Compensation Coefficients	$F_{SensAlgn}$, $\kappa_{SystScal/Mis}$, $\delta\underline{\omega}_{SensBias}$, $\underline{\kappa}_{SystBias}$	$\underline{K}_{Bias}$	(8.1.1.1.1-12)
Nominal Angular Rate Sensor Pulse Output Turn-Around Dead-Band	Constant	db_ω	—
Angular Rate Sensor Quantization Compensation Term (Based On Turn-Around Dead-Band Effect)	$\underline{\alpha}'_m$, Ω_{Wt}, db_ω	$\delta\underline{\alpha}_{QuantC_m}$	(8.1.3.3-7)
Attitude Update Time Interval (Constant)	Constant	T_m	—
Fully Compensated Integrated B Frame Angular Rate Increments	$\underline{\alpha}'_m$, K_{Mis}, $\underline{K}_{Bias}$, $\delta\underline{\alpha}_{QuantC_m}$, T_m	$\underline{\alpha}_m$	(8.1.2.1-4)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Accelerometer Calibration Data	Sensor And System Level Test Results. May Include Active Temperature Functional Dependency.	$G_{SensScalLin}$, $G_{SensScalAsym}$, $G_{SensAlgn}$, $\delta \underline{a}_{SensBias}$, $\lambda_{SystScalLin/Mis}$, $\lambda_{SystScalAsym}$, $\underline{\lambda}_{SystBias}$	—
Nominal Accelerometer Scale Factor (Constant)	Constant	A_{Wt_0}	—
Accelerometer Scale Factor Matrices	A_{Wt_0}, $G_{SensScalLin}$, $G_{SensScalAsym}$, $\lambda_{SystScalLin/Mis}$, $\lambda_{SystScalAsym}$	A_{Wt_+}, A_{Wt_-}	(8.1.1.3-20)
Accelerometer Alignment Compensation Term	$G_{SensAlgn}$, $\lambda_{SystScalLin/Mis}$	G_{Algn}	(8.1.1.3-23), (8.1.1.2.1-16)
Accelerometer Alignment Compensation Coefficients	G_{Algn}	L_{Mis}	(8.1.1.2-6)
Accelerometer Bias Compensation Coefficient	$\delta \underline{a}_{SensBias}$, $\underline{\lambda}_{SystBias}$	$\underline{L}_{Bias}$	(8.1.1.2.1-17)
Accelerometer Quantization Compensation Terms (Based On Measured Pulse Count Residuals)	$\underline{\upsilon}_{CntRes_m}$, A_{Wt_+}, A_{Wt_-}	$\delta \underline{\upsilon}_{QuantC_m}$	(8.1.3.3-10), (8.1.3.3-3)
Incremental Angular Rate Sensor Quantization Compensation Term (Based On Turn-Around Dead-Band Effect)	$\Delta \underline{\alpha}'_m$, $\Delta \underline{\alpha}'_{(l\,:\,m)\,-1}$, Ω_{Wt}, db_ω	$\delta \underline{\alpha}_{QuantC_{l\,:\,m}}$	(8.1.3.3-8)
Accelerometer Size Effect And Anisoinertia Input Parameters Compensated For Scale Factor And Quantization	$\Delta \underline{\alpha}'_m$, $\delta \underline{\alpha}_{QuantC_{l\,:\,m}}$	$\Delta \alpha'^{Q}_{i_m}$	(8.1.4.1.4-10)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Accelerometer Size Effect Compensation Rate And Lever Arms (Constants)	Constant	f_{Size}, $\underline{l}_k$	—
Accelerometer Size Effect Compensation Term Components	$\Delta\underline{\alpha}'_m = \Delta\underline{\alpha}'^{Q}_m$, η_{ij_m}, K_{Mis}, L_{Mis}, f_{Size}, $\underline{l}_k$	$\delta\underline{\upsilon}'_{SizeC_m}$, $\Delta\delta\underline{\upsilon}_{SizeC_m}$,	(8.1.4.1.1.1-15), (8.1.4.1.1.2-3), (8.1.4.1.1.2-5)
Accelerometer Size Effect Compensation Terms	$\delta\underline{\upsilon}'_{SizeC_m}$, $\Delta\delta\underline{\upsilon}_{SizeC_m}$, η_{ij_m}, $\underline{\alpha}_m$, $\Delta\underline{\alpha}'_m = \Delta\underline{\alpha}'^{Q}_m$, f_{Size}, $\underline{l}_k$	$\delta\underline{\upsilon}_{SizeC_m}$, $\delta\underline{v}_{Scul\text{-}SizeC_m}$	(8.1.4.1.1-11), (8.1.4.1.2-13)
Accelerometer Anisoinertia Coefficient (Constant)	Constant	K_{Aniso}	—
Accelerometer Anisoinertia Compensation Term	η_{ij_m}, K_{Aniso}, f_{Size}	$\delta\underline{\upsilon}_{AnisoC_m}$	(8.1.4.2-3)
Fully Compensated Integrated B Frame Acceleration Increments	$\underline{\upsilon}'_m$, L_{Mis}, $\underline{L}_{Bias}$, $\delta\underline{\upsilon}_{SizeC_m}$, $\delta\underline{\upsilon}_{AnisoC_m}$, $\delta\underline{\upsilon}_{QuantC_m}$, T_m	$\underline{\upsilon}_m$	(8.1.2.2-4)
Coning Compensation Coefficient Matrix	K_{Mis}	$K_{MisCone}$	(8.2.1.1-9)
Compensated Coning Increment	$\underline{\beta}'_m$, $K_{MisCone}$	$\underline{\beta}_m$	(8.2.1.1-20)
Sculling & Scrolling Compensation Coefficient Matrices	K_{Mis}, L_{Mis}	LK_1, LK_2, LK_3	(8.2.2.1-31)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Compensated Sculling Increments	$\Delta \underline{v}'_{Scul_{1m}}$, $\Delta \underline{v}'_{Scul_{2m}}$, $\Delta \underline{v}'_{Scul_{3m}}$, LK_1, LK_2, LK_3	$\Delta \underline{v}_{Scul_m}$	(8.2.2.1-41)
Compensated Doubly Integrated B Frame Angular Rate And Acceleration Increments (For High Resolution Position Algorithm)	$\underline{S}'_{\alpha_m}$, K_{Mis}, $\underline{K}_{Bias}$, $\delta\underline{\alpha}_{QuantC_m}$, $\underline{S}'_{\upsilon_m}$, L_{Mis}, $\underline{L}_{Bias}$, $\delta\underline{\upsilon}_{SizeC_m}$, $\delta\underline{\upsilon}_{AnisoC_m}$, $\delta\underline{\upsilon}_{QuantC_m}$, T_m	$\underline{S}_{\alpha_m}$, $\underline{S}_{\upsilon_m}$	(8.1.2.1-6), (8.1.2.2-6)
Compensated Scrolling Increment (For High Resolution Position Algorithm)	$\Delta \underline{R}'_{Scrl_{1m}}$, $\Delta \underline{R}'_{Scrl_{2m}}$, $\Delta \underline{R}'_{Scrl_{3m}}$, LK_1, LK_2, LK_3	$\Delta \underline{R}_{Scrl_m}$	(8.2.3.1-9)

<u>EARTH RELATED PARAMETERS</u> — Table 7.5-1 Earth Related Parameter Calculations

<u>VELOCITY CALCULATIONS</u>

Table 7.5-1 Velocity Calculations Up To B Frame Integrated Specific Force Acceleration Increment — Table 7.5-1 Velocity Calculations Up To B Frame Integrated Specific Force Acceleration Increment

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
B Frame Integrated Specific Force Acceleration Increment	$\underline{\upsilon}_m$, $\Delta\underline{v}_{Rot_m}$, $\Delta\underline{v}_{Scul_m}$, $\delta\underline{v}_{Scul\text{-}SizeC_m}$	$\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$	(8.1.4.1-14)
Remaining Velocity Calculations	Table 7.5-1 Velocity Calculations Following B Frame Integrated Specific Force Acceleration Increment		
POSITION CALCULATIONS			
Table 7.5-1 Position Calculations Up To Body Frame Position Increment Due To Specific Force Acceleration (High Resolution Position Algorithm)	Table 7.5-1 Position Calculations Up To Body Frame Position Increment Due To Specific Force Acceleration (High Resolution Position Algorithm)		
Body Frame Position Increment Due To Specific Force Acceleration (High Resolution Position Algorithm)	$\underline{S}_{\upsilon_m}$, $\Delta\underline{R}_{Rot_m}$, $\Delta\underline{R}_{Scrl_m}$, $\delta\underline{v}_{Scul\text{-}SizeC_m}$, T_m	$\Delta\underline{R}_{SF_m}^{B}$	(8.1.4.1-16)
Remaining Position Calculations	Table 7.5-1 Position Calculations Following Body Frame Position Increment Due To Specific Force Acceleration (High Resolution Position Algorithm)		
ATTITUDE CALCULATIONS			
Table 7.5-1 Attitude Calculations Up To Roll, Pitch, True Heading Euler Angle Outputs	Table 7.5-1 Attitude Calculations Up To Roll, Pitch, True Heading Euler Angle Outputs		
Sensor Assembly Attitude Alignment Calibration Coefficients And Mount Geometry	System Level Test Results And Geometry	$\underline{J}$, C_{VRF}^{M}	—
Attitude Compensation For Sensor Assembly Misalignment and Mounting	C_B^L, C_{VRF}^M, $\underline{J}$	C_{VRF}^{L}	(8.3-1), (8.3-2)

ALGORITHM FUNCTION	INPUT	OUTPUT	EQUATION
Roll, Pitch, True Heading Euler Angle Outputs	C_{VRF}^{L}, α	ϕ, θ, ψ_{True}	(4.1.2-1), (4.1.2-2) With C_{VRF}^{L} Interpreted As C_{B}^{L}

9 Sensor Assembly Jitter Compensation

9.0 OVERVIEW

This chapter derives computational routines for removing sensor assembly jitter motion from the computed navigation parameters in a strapdown inertial navigation system. For this development, jitter is defined as angular and linear movement of the INS strapdown sensor assembly relative to "rigid" user vehicle reference axes. Jitter is produced by vehicle dynamic disturbances that create vehicle bending, and that also excite sensor assembly motion on its isolators (if used) within the INS. Additionally, for sensor assemblies using mechanically dithered ring laser gyro rate sensors, jitter is produced by the sensor assembly reaction to gyro dither torque.

Three basic methods might be considered for jitter removal; 1. Filtering of inertial sensor data prior to integration into navigation data (attitude, velocity, position), 2. Filtering of the inertially computed navigation data, and 3. Inertially determining the jitter and subtracting it from the computed navigation data. The first method also removes real coning/sculling/scrolling motion from the inertial sensor output, hence, is only practical when it is known that these effects are not significant for the particular application requirements. The second method can distort the dynamic characteristics of the desired jitter-free "rigid" user vehicle navigation data since it is also filtered as part of the jitter removal process. The third jitter subtraction method removes the jitter while preserving the dynamic characteristics of the jitter-free solution, and will be the method described in this chapter. The success of the jitter subtraction process depends on the accuracy for determining jitter motion based on inertial sensor outputs.

In this chapter we will first develop an analytical definition for jitter in attitude, velocity and position. Routines for calculating jitter follow from the analytical definition. Finally, equations will be developed for removing the jitter from the basic attitude, velocity and position navigation parameters.

The principal coordinate frames utilized in this chapter are the B, L, N and E Frames as defined in Section 2.2.

9.1 ANALYTICAL DESCRIPTION OF JITTER

The analytical description of jitter is developed by separately defining the motion of the strapdown sensor assembly and the user vehicle, and then taking the difference to define the jitter components. The general differential equations that describe the motion of the sensor assembly have been derived in Chapter 4 as Equations (4.1-1), (4.3-18) and (4.3-1) summarized below:

$$\dot{C}_B^L = C_B^L \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IL}^L \times\right) C_B^L$$

$$\dot{\underline{v}}^N = \underline{a}_{SF}^N + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N \tag{9.1-1}$$

$$\dot{\underline{R}}^E = \underline{v}^E$$

where

$\underline{\omega}_{IB}$ = Angular rotation rate vector of the strapdown sensor assembly body B Frame relative to non-rotating inertial space, sensed in the B Frame by the strapdown angular rate sensors.

$\underline{\omega}_{IL}$ = Angular rotation rate vector of the locally level attitude reference L Frame relative to non-rotating inertial space.

C_B^L = Direction cosine matrix that transforms vectors from B to L Frame coordinates.

$\underline{v}$ = Velocity vector relative to the earth.

$\underline{\omega}_{IE}$ = Earth E Frame rotation rate vector relative to non-rotating inertial space.

$\underline{\omega}_{EN}$ = Angular rate vector of the locally level navigation N Frame relative to the E Frame.

$\underline{a}_{SF}$ = Specific force acceleration vector of the sensor assembly, sensed in the B Frame by the strapdown accelerometers.

$\underline{g}_P$ = Plumb-bob gravity vector.

$\underline{R}$ = Sensor assembly position distance vector from earth's center.

An equivalent equation for $\dot{\underline{R}}^E$ in (9.1-1) can be developed in the N-Frame from (4.3-1), (4.3-2) and generalized Equation (3.4-6):

$$\dot{\underline{R}}^N = C_E^N\, \dot{\underline{R}}^E + \underline{\omega}_{NE}^N \times \underline{R}^N = \underline{v}^N - \underline{\omega}_{EN}^N \times \underline{R}^N \tag{9.1-2}$$

Equations (9.1-1) with (9.1-2) are:

$$\dot{C} = C\left(\underline{\omega}_B^B \times\right) - \left(\underline{\omega}_{IL}^L \times\right) C$$

$$\dot{\underline{v}}^N = \underline{a}_{SF}^N + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N \tag{9.1-3}$$

$$\dot{\underline{R}}^N = \underline{v}^N - \underline{\omega}_{EN}^N \times \underline{R}^N$$

where for notation simplicity in this development we have defined:

$$C \equiv C_B^L \qquad \underline{\omega}_B^B \equiv \underline{\omega}_{IB}^B$$

Equations (9.1-3) describe the motion of the sensor assembly. An equivalent set of equations can be written that describe the motion of the user vehicle reference frame (i.e., without jitter):

$$\dot{C}_{UV} = C_{UV}\left(\underline{\omega}_{UV}^{UV} \times\right) - \left(\underline{\omega}_{IL}^L \times\right)_{UV} C_{UV}$$

$$\dot{\underline{v}}_{UV}^N = \underline{a}_{SF_{UV}}^N + \underline{g}_{P_{UV}}^N - \left[\left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N\right]_{UV} \tag{9.1-4}$$

$$\dot{\underline{R}}_{UV}^N = \underline{v}_{UV}^N - \left(\underline{\omega}_{EN}^N \times \underline{R}^N\right)_{UV}$$

where

UV = User vehicle reference frame (without jitter). If the sensor assembly was free of angular jitter, the UV and B Frames would be parallel.

$(\,)^{UV}$ = Designates vector components evaluated along UV Frame axes.

$(\,)_{UV}$ = Designates parameters that would be calculated in a strapdown INS that is free of jitter. In the absence of jitter, the $(\,)_{UV}$ and actual inertial sensor assembly parameters would be identical.

$\underline{\omega}_{UV}$, $\underline{a}_{SF_{UV}}$ = Values for $\underline{\omega}_B$, $\underline{a}_{SF}$ if the inertial sensor assembly was jitter free.

We now define the sensor assembly jitter in terms of the variation between the Equation (9.1-3) and (9.1-4) navigation parameters:

$$\Delta\underline{R}_{JTR}^N \equiv \underline{R}^N - \underline{R}_{UV}^N \qquad \Delta\underline{v}_{JTR}^N \equiv \underline{v}^N - \underline{v}_{UV}^N \qquad \left(\Delta\underline{\theta}_{JTR}^B \times\right) \equiv C_B^{UV} - I \tag{9.1-5}$$

where

$\Delta\underline{v}_{JTR}^N$, $\Delta\underline{R}_{JTR}^N$ = Velocity and position jitter motion of the sensor assembly as described in the N Frame.

$\Delta\underline{\theta}^{B}_{JTR}$ = Angular jitter rotation vector of the B Frame relative to the UV Frame as described in the B Frame.

C^{UV}_{B} = Direction cosine matrix that transforms vectors from the B Frame to the UV Frame.

The $\Delta\underline{\theta}^{B}_{JTR}$ expression in Equations (9.1-5) is based on a first order approximation to general rotation vector Equation (3.2.2.1-8).

A differential equation in the B Frame for the velocity jitter rate vector is developed by first defining:

$$\Delta\underline{v}^{L}_{JTR} = C^{L}_{N}\, \Delta\underline{v}^{N}_{JTR} \qquad \Delta\underline{v}^{B}_{JTR} = C^{T}\, \Delta\underline{v}^{L}_{JTR} \tag{9.1-6}$$

Taking the derivative of Equations (9.1-6) and recognizing from the Section 2.2 definitions that Frames N and L are parallel, hence, their relative attitude is constant, obtains:

$$\Delta\dot{\underline{v}}^{L}_{JTR} = C^{L}_{N}\, \Delta\dot{\underline{v}}^{N}_{JTR} \qquad \Delta\dot{\underline{v}}^{B}_{JTR} = \dot{C}^{T}\, \Delta\underline{v}^{L}_{JTR} + C^{T}\, \Delta\dot{\underline{v}}^{L}_{JTR} \tag{9.1-7}$$

or, with (9.1-5),

$$\Delta\dot{\underline{v}}^{B}_{JTR} = \dot{C}^{T}\, \Delta\underline{v}^{L}_{JTR} + C^{T}\, C^{L}_{N} \left(\dot{\underline{v}}^{N} - \dot{\underline{v}}^{N}_{UV}\right) \tag{9.1-8}$$

The transpose of $\dot{C}$ from (9.1-3) coupled with $\dot{\underline{v}}^{N}$ and $\dot{\underline{v}}^{N}_{UV}$ from (9.1-3) and (9.1-4), when substituted in (9.1-8), yields:

$$\begin{aligned} \Delta\dot{\underline{v}}^{B}_{JTR} = & \left[- \left(\underline{\omega}^{B}_{B}\times\right) C^{T} + C^{T} \left(\underline{\omega}^{L}_{IL}\times\right)\right] \Delta\underline{v}^{L}_{JTR} \\ & + C^{T}\, C^{L}_{N} \left(\underline{a}^{N}_{SF} - \underline{a}^{N}_{SF_{UV}} + \underline{a}^{N}_{G/COR} - \underline{a}^{N}_{G/COR_{UV}}\right) \end{aligned} \tag{9.1-9}$$

with

$$\begin{aligned} \underline{a}^{N}_{G/COR} &\equiv \underline{g}^{N}_{P} - \left(\underline{\omega}^{N}_{EN} + 2\, \underline{\omega}^{N}_{IE}\right) \times \underline{v}^{N} \\ \underline{a}^{N}_{G/COR_{UV}} &\equiv \underline{g}^{N}_{P_{UV}} - \left[\left(\underline{\omega}^{N}_{EN} + 2\, \underline{\omega}^{N}_{IE}\right) \times \underline{v}^{N}\right]_{UV} \end{aligned} \tag{9.1-10}$$

Since the UV and B Frames are nearly at the same position, the $\underline{g}^{N}_{P}$ and $\underline{g}^{N}_{P_{UV}}$ terms in (9.1-10) are virtually identical. We also note that the $\left(\underline{\omega}^{N}_{EN} + 2\, \underline{\omega}^{N}_{IE}\right) \times \underline{v}^{N}$ term in (9.1-10) is very close to the "UV" version, and that both are generally small compared to $\underline{a}^{N}_{SF}$ and $\underline{a}^{N}_{SF_{UV}}$.

This provides the justification for neglecting the difference between $\underline{a}^N_{G/COR}$ and $\underline{a}^N_{G/COR_{UV}}$ in (9.1-9). We also note that $\underline{\omega}^L_{IL}$ is generally much smaller than $\underline{\omega}^B_B$, hence, this term can also be neglected in (9.1-9). Applying these simplifications to (9.1-9) with (9.1-6), yields the approximate form:

$$\begin{aligned} \Delta\dot{\underline{v}}^B_{JTR} &\approx C^T C^L_N \left(\underline{a}^N_{SF} - \underline{a}^N_{SF_{UV}}\right) - \left(\underline{\omega}^B_B \times\right) C^T \Delta\underline{v}^L_{JTR} \\ &= C^T C^L_N \left(\underline{a}^N_{SF} - \underline{a}^N_{SF_{UV}}\right) - \underline{\omega}^B_B \times \Delta\underline{v}^B_{JTR} \end{aligned} \tag{9.1-11}$$

We now define:

$$\Delta\underline{a}^N_{JTR} \equiv \underline{a}^N_{SF} - \underline{a}^N_{SF_{UV}} \qquad \Delta\underline{a}^B_{JTR} = C^T C^L_N \Delta\underline{a}^N_{JTR} \tag{9.1-12}$$

where

$\Delta\underline{a}^B_{JTR}, \Delta\underline{a}^N_{JTR}$ = Jitter specific force acceleration as described in the B and N Frames.

With (9.1-12), Equation (9.1-11) becomes:

$$\Delta\dot{\underline{v}}^B_{JTR} = \Delta\underline{a}^B_{JTR} - \underline{\omega}^B_B \times \Delta\underline{v}^B_{JTR} \tag{9.1-13}$$

Equation (9.1-13) includes approximations that are small, but that over time, could integrate into a sizable error in $\Delta\underline{v}^B_{JTR}$. To preclude the long term error build-up, a first order feedback term is added to (9.1-13) to obtain the final result:

$$\Delta\dot{\underline{v}}^B_{JTR} = \Delta\underline{a}^B_{JTR} - \underline{\omega}^B_B \times \Delta\underline{v}^B_{JTR} - C_{JTR}\, \Delta\underline{v}^B_{JTR} \tag{9.1-14}$$

where

C_{JTR} = Drift control feedback gain.

The C_{JTR} coefficient is set to a low enough value to avoid corruption of the high frequency dynamic characteristics of $\Delta\underline{v}^B_{JTR}$.

A similar procedure is used to develop the $\Delta\underline{R}^B_{JTR}$ rate equation. First:

$$\Delta\underline{R}^B_{JTR} = C^T C^L_N \Delta\underline{R}^N_{JTR} \tag{9.1-15}$$

Then:

$$
\begin{aligned}
\Delta \underline{\dot{R}}^{B}_{JTR} &= \dot{C}^{T}\, C^{L}_{N}\, \Delta \underline{R}^{N}_{JTR} + C^{T}\, C^{L}_{N}\, \Delta \underline{\dot{R}}^{N}_{JTR} \\
&= \dot{C}^{T}\, \Delta \underline{R}^{L}_{JTR} + C^{T}\, C^{L}_{N} \left(\underline{\dot{R}}^{N} - \underline{\dot{R}}^{N}_{UV} \right)
\end{aligned}
\qquad (9.1\text{-}16)
$$

With (9.1-3) and (9.1-4) we obtain:

$$
\begin{aligned}
\Delta \underline{\dot{R}}^{B}_{JTR} &= \left[- \left(\underline{\omega}^{B}_{B} \times \right) C^{T} + C^{T} \left(\underline{\omega}^{L}_{IL} \times \right) \right] \Delta \underline{R}^{L}_{JTR} \\
&\qquad + C^{T}\, C^{L}_{N} \left[\underline{v}^{N} - \underline{v}^{N}_{UV} - \underline{\omega}^{N}_{EN} \times \underline{R}^{N} + \left(\underline{\omega}^{N}_{EN} \times \underline{R}^{N} \right)_{UV} \right] \\
&\approx - \left(\underline{\omega}^{B}_{B} \times \right) C^{T}\, \Delta \underline{R}^{L}_{JTR} + C^{T}\, C^{L}_{N} \left(\underline{v}^{N} - \underline{v}^{N}_{UV} \right)
\end{aligned}
\qquad (9.1\text{-}17)
$$

With (9.1-5), (9.1-6), (9.1-15) and feedback drift control, Equation (9.1-17) assumes the final form:

$$
\Delta \underline{\dot{R}}^{B}_{JTR} = \Delta \underline{v}^{B}_{JTR} - \underline{\omega}^{B}_{B} \times \Delta \underline{R}^{B}_{JTR} - C_{JTR}\, \Delta \underline{R}^{B}_{JTR} \qquad (9.1\text{-}18)
$$

It remains to develop a differential equation for the rate of change of $\Delta \underline{\theta}^{B}_{JTR}$ in (9.1-5). The derivation begins by first noting that by their definitions,

$$
C = C_{UV}\, C^{UV}_{B} \qquad (9.1\text{-}19)
$$

The derivative of (9.1-19) is:

$$
\dot{C} = \dot{C}_{UV}\, C^{UV}_{B} + C_{UV}\, \dot{C}^{UV}_{B} \qquad (9.1\text{-}20)
$$

From (9.1-5) we can write:

$$
\dot{C}^{UV}_{B} = \left(\Delta \underline{\dot{\theta}}^{B}_{JTR} \times \right) \qquad (9.1\text{-}21)
$$

From (9.1-3), (9.1-4) and (9.1-19) with approximations and generalized Equation (3.1.1-40):

$$
\dot{C} = C \left(\underline{\omega}^{B}_{B} \times \right) \qquad (9.1\text{-}22)
$$

$$
\dot{C}_{UV} = C_{UV} \left(\underline{\omega}^{UV}_{UV} \times \right) = C_{UV}\, C^{UV}_{B} \left(\underline{\omega}^{B}_{UV} \times \right) C^{B}_{UV} = C \left(\underline{\omega}^{B}_{UV} \times \right) C^{B}_{UV} \qquad (9.1\text{-}23)
$$

Substituting (9.1-19) and (9.1-21) - (9.1-23) into (9.1-20) obtains:

$$
\begin{aligned}
C\left(\underline{\omega}_{B}^{B}\times\right) &= C\left(\underline{\omega}_{UV}^{B}\times\right) C_{UV}^{B}\; C_{B}^{UV} + C_{UV}\left(\Delta\underline{\dot{\theta}}_{JTR}^{B}\times\right) \\
&= C\left(\underline{\omega}_{UV}^{B}\times\right) + C\, C_{UV}^{B}\left(\Delta\underline{\dot{\theta}}_{JTR}^{B}\times\right)
\end{aligned}
\tag{9.1-24}
$$

or

$$
\left(\Delta\underline{\dot{\theta}}_{JTR}^{B}\times\right) = C_{B}^{UV}\left[\left(\underline{\omega}_{B}^{B} - \underline{\omega}_{UV}^{B}\right)\times\right] \approx \left[\left(\underline{\omega}_{B}^{B} - \underline{\omega}_{UV}^{B}\right)\times\right]
\tag{9.1-25}
$$

or

$$
\Delta\underline{\dot{\theta}}_{JTR}^{B} = \underline{\omega}_{B}^{B} - \underline{\omega}_{UV}^{B}
\tag{9.1-26}
$$

We define:

$$
\Delta\underline{\omega}_{JTR}^{B} \equiv \underline{\omega}_{B}^{B} - \underline{\omega}_{UV}^{B}
\tag{9.1-27}
$$

where

$\Delta\underline{\omega}_{JTR}^{B}$ = Jitter angular rate as described in the B Frame.

Finally, with (9.1-27) and introduction of low frequency drift control, Equation (9.1-26) assumes the final form:

$$
\Delta\underline{\dot{\theta}}_{JTR}^{B} = \Delta\underline{\omega}_{JTR}^{B} - C_{JTR}\,\Delta\underline{\theta}_{JTR}^{B}
\tag{9.1-28}
$$

In summary, Equations (9.1-14), (9.1-18), and (9.1-28) with (9.1-5), (9.1-6), and (9.1-15) define the differential equations for jitter motion of the B Frame relative to the UV Frame. These equations are repeated below:

$$
\begin{aligned}
&\left(\Delta\underline{\theta}_{JTR}^{B}\times\right) \equiv C_{B}^{UV} - I \\
&\Delta\underline{v}_{JTR}^{N} \equiv \underline{v}^{N} - \underline{v}_{UV}^{N} \qquad \Delta\underline{v}_{JTR}^{B} = C^{T}\, C_{N}^{L}\, \Delta\underline{v}_{JTR}^{N} \\
&\Delta\underline{R}_{JTR}^{N} \equiv \underline{R}^{N} - \underline{R}_{UV}^{N} \qquad \Delta\underline{R}_{JTR}^{B} = C^{T}\, C_{N}^{L}\, \Delta\underline{R}_{JTR}^{N}
\end{aligned}
\tag{9.1-29}
$$

$$
\begin{aligned}
\Delta\underline{\dot{\theta}}_{JTR}^{B} &= \Delta\underline{\omega}_{JTR}^{B} - C_{JTR}\,\Delta\underline{\theta}_{JTR}^{B} \\
\Delta\underline{\dot{v}}_{JTR}^{B} &= \Delta\underline{a}_{JTR}^{B} - \underline{\omega}_{B}^{B}\times\Delta\underline{v}_{JTR}^{B} - C_{JTR}\,\Delta\underline{v}_{JTR}^{B} \\
\Delta\underline{\dot{R}}_{JTR}^{B} &= \Delta\underline{v}_{JTR}^{B} - \underline{\omega}_{B}^{B}\times\Delta\underline{R}_{JTR}^{B} - C_{JTR}\,\Delta\underline{R}_{JTR}^{B}
\end{aligned}
\tag{9.1-30}
$$

9.2 JITTER RATE/ACCELERATION MEASUREMENT

The $\Delta\underline{\omega}_{JTR}^{B}$ and $\Delta\underline{a}_{JTR}^{B}$ angular-rate/linear-acceleration jitter terms in Equations (9.1-30) can be evaluated from the strapdown INS angular rate sensor and accelerometer signals by employing a suitable filter. This method is based on the assumption that the jitter motion can be defined to be that portion of the total sensor output signal with frequency content above a specified cutoff frequency. Sensor outputs with frequency content below the cutoff then represent the jitter free (UV) motion. The basic structure of the jitter filter is a low pass stage that measures the jitter free (UV) motion, followed by subtraction of the UV motion from the full sensor output to determine the jitter signal. The subtraction process includes compensation for the dynamic delay time in the jitter filter. Analytically then, the overall method for determining angular rate and acceleration jitter is as follows:

$$
\begin{aligned}
&\underline{\omega}_{UV_F}^{B} = F_{LP}\left(\underline{\omega}_{B}^{B}\right) && \underline{a}_{SF/UV_F}^{B} = F_{LP}\left(\underline{a}_{SF}^{B}\right) \\
&\underline{\omega}_{B_{DL}}^{B} = G_{DL}\left(\underline{\omega}_{B}^{B}\right) && \underline{a}_{SF_{DL}}^{B} = G_{DL}\left(\underline{a}_{SF}^{B}\right) \\
&\Delta\underline{\omega}_{JTR_{DL}}^{B} = \underline{\omega}_{B_{DL}}^{B} - \underline{\omega}_{UV_F}^{B} && \Delta\underline{a}_{JTR_{DL}}^{B} = \underline{a}_{SF_{DL}}^{B} - \underline{a}_{SF/UV_F}^{B}
\end{aligned}
\tag{9.2-1}
$$

where

$F_{LP}(\;)$ = Transfer function of the low pass filter used to estimate angular-rate/linear-acceleration jitter from the sensed B frame rate/acceleration data.

$G_{DL}(\;)$ = Delay function to compensate for dynamic time delay associated with the $F_{LP}(\;)$ filter.

F = Subscript designating $F_{LP}(\;)$ filter output.

DL = Subscript designation for time delayed signal in which delay time equals the $F_{LP}(\;)$ filter dynamic delay time.

The $F_{LP}(\;)$ function in (9.2-1) should be selected to meet a specified cutoff frequency and attenuation characteristic. It is advantageous if the selected filter is implemented with a fixed dynamic delay time characteristic for all input frequencies (i.e., linear phase versus frequency response) such as exhibited by a Remez digital filter. This allows the $G_{DL}(\;)$ function to be implemented as a pure time delay to perfectly compensate for filter delay time match-up error. For $F_{LP}(\;)$ with a fixed time delay, the $G_{DL}(\;)$ function would represent the value of its argument at the $F_{LP}(\;)$ filter input time. Note, that the $\Delta\underline{\omega}_{JTR}^{B}$ and $\Delta\underline{a}_{JTR}^{B}$ terms calculated per Equations (9.2-1) will then also contain a time delay (as indicated by "DL").

The $\Delta\underline{\omega}^{B}_{JTR_{DL}}$, $\Delta\underline{a}^{B}_{JTR_{DL}}$ terms calculated in (9.2-1) are used to compute the $\Delta\underline{\theta}^{B}_{JTR}$, $\Delta\underline{v}^{B}_{JTR}$, and $\Delta\underline{R}^{B}_{JTR}$ jitter parameters by integrating Equations (9.1-30) with $\Delta\underline{\omega}^{B}_{JTR_{DL}}$, $\Delta\underline{a}^{B}_{JTR_{DL}}$ as input. Since, as discussed above, the $\Delta\underline{\omega}^{B}_{JTR_{DL}}$, $\Delta\underline{a}^{B}_{JTR_{DL}}$ inputs to (9.1-30) will be time delayed, their use in (9.1-30) will produce the same time delay in the $\Delta\underline{\theta}^{B}_{JTR}$, $\Delta\underline{v}^{B}_{JTR}$, and $\Delta\underline{R}^{B}_{JTR}$ outputs. The revised form of (9.1-30) that acknowledges the time delay factor is as follows:

$$
\begin{aligned}
\Delta\dot{\underline{\theta}}^{B}_{JTR_{DL}} &= \Delta\underline{\omega}^{B}_{JTR_{DL}} - C_{JTR}\, \Delta\underline{\theta}^{B}_{JTR_{DL}} \\
\Delta\dot{\underline{v}}^{B}_{JTR_{DL}} &= \Delta\underline{a}^{B}_{JTR_{DL}} - \underline{\omega}^{B}_{B_{DL}} \times \Delta\underline{v}^{B}_{JTR_{DL}} - C_{JTR}\, \Delta\underline{v}^{B}_{JTR_{DL}} \qquad (9.2\text{-}2) \\
\Delta\dot{\underline{R}}^{B}_{JTR_{DL}} &= \Delta\underline{v}^{B}_{JTR_{DL}} - \underline{\omega}^{B}_{B_{DL}} \times \Delta\underline{R}^{B}_{JTR_{DL}} - C_{JTR}\, \Delta\underline{R}^{B}_{JTR_{DL}}
\end{aligned}
$$

The integral of Equations (9.2-2) with input from (9.2-1) provides the desired measurements of sensor assembly jitter to be removed from the inertially computed attitude, velocity, position navigation parameters.

9.3 JITTER FILTER INPUTS

The $\underline{\omega}^{B}_{B}$ and $\underline{a}^{B}_{SF}$ angular-rate-sensor/accelerometer inputs to the Equation (9.2-1) jitter filters can be approximated by the outputs from the inertial sensors compensated for only scale factor as in Equations (8.1.1.1-8) and (8.1.1.2-8):

$$
\underline{\omega}^{B}_{B} \approx \Omega_{Wt}\, \underline{\omega}_{Puls} \qquad \underline{a}^{B}_{SF} \approx A_{Wt}\, \underline{a}_{SF_{Puls}} \qquad (9.3\text{-}1)
$$

where

Ω_{Wt} = Angular rate sensor triad scale factor pulse weighting matrix (radians per pulse).

$\underline{\omega}_{Puls}$ = Uncompensated angular rate sensor triad “instantaneous” output pulse rate vector (pulses per sec).

A_{Wt} = Accelerometer triad scale factor pulse weighting matrix (ft per sec per pulse).

$\underline{a}_{SF_{Puls}}$ = Uncompensated accelerometer triad “instantaneous” output pulse rate vector (pulses per sec).

For improved performance, the accelerometer data can also be compensated for accelerometer size effect prior to jitter filter input. From Equation (8.1.4.1-9) (with G_{Algn} approximated by identity), the accelerometer size effect correction for each accelerometer would classically be given by:

$$\delta a_{Size_k} = \left[\dot{\underline{\omega}}_B^B \times \underline{l}_k^B + \underline{\omega}_B^B \times \left(\underline{\omega}_B^B \times \underline{l}_k^B \right) \right] \cdot \underline{u}_k^B \tag{9.3-2}$$

where

$\underline{u}_k$ = Unit vector along the accelerometer k input axis.

$\underline{l}_k$ = Lever arm from the accelerometer k center of seismic mass to the sensor assembly navigation reference point.

δa_{Size_k} = Size effect correction to accelerometer k output.

For normal application conditions, the δa_{Size_k} correction given by Equation (9.3-2) would be small and not significant enough to be included in the calculation of sensor assembly jitter motion. However, under test laboratory conditions when jitter may be introduced through angular oscillations of the INS on a test rotation table, the angular acceleration term may no longer be negligible. To avoid potential performance problems under laboratory jitter test conditions, the $\dot{\underline{\omega}}_B^B \times \underline{l}_k^B$ angular acceleration term in Equation (9.3-2) can be applied to the $\underline{a}_{SF}^B$ signal prior to processing by the Equation (9.2-1) jitter filters. The $\dot{\underline{\omega}}_B^B$ term in (9.3-2) would be calculated as the rate of change $\underline{\omega}_B^B$ from Equation (9.3-1). Digital algorithms for the above operations can be derived as in Section 8.1.4.1 and its subsections.

9.4 JITTER REMOVAL

After the jitter motion parameters $\Delta\underline{\theta}_{JTR_{DL}}^B$, $\Delta\underline{v}_{JTR_{DL}}^B$, $\Delta\underline{R}_{JTR_{DL}}^B$ are calculated by integrating Equations (9.2-2), they can be used to remove jitter effects from the INS navigation parameters. Jitter can also be removed from the B Frame angular rate and acceleration signals for output purposes using $\Delta\underline{\omega}_{JTR_{DL}}^B$, $\Delta\underline{a}_{JTR_{DL}}^B$ from (9.2-1). The means for jitter removal is through the inverse of (9.1-5), (9.1-6), (9.1-15), and (9.1-19). Since the calculated jitter parameters include a time delay, INS navigation parameters must also be delayed by the same amount so that the jitter corrections can be accurately synchronized. An additional delay must typically be included so that the final result is synchronized with navigation parameter update cycle times. Assuming identical update rates for the navigation parameters, the net result for the angular rate, acceleration, attitude and velocity parameters would be as follows:

$$\Delta\underline{\omega}^B_{JTR_{SDL}} = G_{S/DL}\left(\Delta\underline{\omega}^B_{JTR_{DL}}\right) \qquad \Delta\underline{a}^B_{JTR_{SDL}} = G_{S/DL}\left(\Delta\underline{a}^B_{JTR_{DL}}\right)$$
$$\Delta\underline{\theta}^B_{JTR_{SDL}} = G_{S/DL}\left(\Delta\underline{\theta}^B_{JTR_{DL}}\right) \qquad \Delta\underline{v}^B_{JTR_{SDL}} = G_{S/DL}\left(\Delta\underline{v}^B_{JTR_{DL}}\right)$$
$$\Delta\underline{v}^N_{JTR_{SDL}} = C^N_L \, C_{SDL} \, \Delta\underline{v}^B_{JTR_{SDL}}$$

$$\underline{\omega}^B_{SDL} = G_{SDL}\left(\underline{\omega}^B_B\right) \qquad \underline{a}^B_{SF_{SDL}} = G_{SDL}\left(\underline{a}^B_{SF}\right) \qquad \text{(9.4-1)}$$
$$C_{SDL} = G_{SDL}(C) \qquad \underline{v}^N_{SDL} = G_{SDL}\left(\underline{v}^N\right)$$

$$\underline{\omega}^B_{UV\,SDL} = \underline{\omega}^B_{SDL} - \Delta\underline{\omega}^B_{JTR_{SDL}} \qquad \underline{a}^B_{SF/UV\,SDL} = \underline{a}^B_{SDL} - \Delta\underline{a}^B_{JTR_{SDL}}$$
$$C_{UV\,SDL} = C_{SDL}\left[I - \left(\Delta\underline{\theta}^B_{JTR_{SDL}} \times\right)\right] \qquad \underline{v}^N_{UV\,SDL} = \underline{v}^N_{SDL} - \Delta\underline{v}^N_{JTR_{SDL}}$$

where

$G_{SDL}(\,)$ = Delay function corresponding to specified integer number of navigation parameter update cycles.

$G_{S/DL}(\,)$ = Delay function applied to $G_{DL}(\,)$ delay filter outputs that results in a net delay (including the $G_{DL}(\,)$ delay time) equal to the specified integer number of navigation parameter update cycles.

SDL = Subscript designation for time delayed signal in which the delay time equals the specified integer number of navigation parameter update cycles.

The C and $\underline{v}^N$ terms in (9.4-1) would be the C^L_B attitude direction cosine matrix and $\underline{v}^N$ earth reference velocity as calculated by the strapdown inertial navigation integration algorithms of Chapter 7 (including the Chapter 8 sensor compensation) and the Chapter 19, Section 19.1 unified algorithms (if applied). As for the $G_{DL}(\,)$ delay function, the associated time delays would be selected so that the $G_{SDL}(\,)$ and $G_{S/DL}(\,)$ functions can be realized as equal to their arguments at previous computer update cycle times corresponding to the time delays.

The vertical component of position jitter can be removed similarly from the altitude parameter:

$$\Delta\underline{R}^B_{JTR_{SDL}} = G_{S/DL}\left(\Delta\underline{R}^B_{JTR_{DL}}\right) \qquad \Delta\underline{R}^N_{JTR_{SDL}} = C^N_L \, C_{SDL} \, \Delta\underline{R}^B_{JTR_{SDL}}$$
$$h_{SDL} = G_{SDL}(h) \qquad h_{UV_{SDL}} = h_{SDL} - \underline{u}^N_{ZN} \cdot \Delta\underline{R}^N_{JTR_{SDL}} \qquad \text{(9.4-2)}$$

where

h = Altitude.

$\underline{u}_{ZN}^{N}$ = Unit vector upward along the Z axis of the N Frame.

The h altitude term in (9.4-2) would be as calculated by the Chapter 7 strapdown inertial navigation integration algorithms.

The horizontal component of position jitter can be removed from the INS navigation data by adjustment of the C_N^E matrix (i.e., the direction cosine matrix that transforms vectors from the N Frame to the E Frame). The method is by defining an equivalent earth surface angular rotation vector corresponding to the calculated position jitter vector, and then removing the horizontal angular rotation components from C_N^E. The method is directly analogous to Equations (7.3.1-6), (7.3.1-8) and (7.3.1-11) for position updating. Using these equations as a template, assuming that position and velocity updating cycle rates are equal (i.e., equivalent to setting the n and m rates equal in the previous referenced equations), approximating F_C^N as the identity matrix divided by the average local radius of curvature with the 3, 3 element set to zero, and carrying only first order terms then obtains:

$$\underline{\xi}_{JTR_{SDL}}^{N} = \frac{1}{R}\left(\underline{u}_{ZN}^{N} \times \Delta\underline{R}_{JTR_{SDL}}^{N}\right) \qquad D_{SDL} = G_{SDL}(D)$$

$$D_{UV_{SDL}} = D_{SDL}\left[I - \left(\underline{\xi}_{JTR_{SDL}}^{N} \times\right)\right] \tag{9.4-3}$$

where

$D = C_N^E$ for simplicity in Equation (9.4-3).

R = Average local radius of curvature which usually can safely be approximated as earth's equatorial radius.

$\underline{\xi}_{JTR_{SDL}}^{N}$ = Horizontal angular jitter of the N Frame.

The D (or C_N^E) position direction cosine matrix in (9.4-3) would be as calculated by the Chapter 7 strapdown inertial navigation integration algorithms (including the Chapter 19, Section 19.1 unified algorithms if applied).

9.5 NAVIGATION OUTPUT PARAMETERS

Once the jitter is removed from the basic attitude, velocity, and position integration parameters as described in Equations (9.4-1) through (9.4-3), traditional navigation output data can be calculated from the jitter free results such as Equations (4.4.2.1-3) for latitude/longitude/wander-angle, Equations (4.3.1-4) for north/east/vertical velocity and Equations (4.1.2-1), (4.1.2-2), (8.3-1) and (8.3-2) for roll/pitch/heading attitude. The altitude output would be $h_{UV_{SDL}}$ directly from Equations (9.4-2). B Frame angular rates and acceleration outputs would be obtained from $\underline{\omega}^{B}_{UV_{SDL}}$, $\underline{a}^{B}_{SF/UV_{SDL}}$ after transformation into the appropriate vehicle coordinate frame using Equation (8.3-3).

10 Vibration Effects Analysis

10.0 OVERVIEW

In this chapter we will develop analytical techniques that can be used to assess strapdown INS performance under dynamic vibration environments. In particular, methods will be developed for assessing strapdown computation algorithm accuracy and inertial sensor vibration sensitive error effects under linear and angular vibration in sinusoidal and random vibration environments. Section 10.1 analyzes the effect of sinusoidal linear and angular vibrations of the strapdown inertial sensor assembly on true attitude/velocity/position, strapdown algorithm computed attitude/velocity/position, and on strapdown inertial sensor error. Section 10.2 provides a brief review of linear system response to sinusoidal and random inputs. Sections 10.3 and 10.4 apply Section 10.2 to translate the Section 10.1 sinusoidal response results into strapdown sensor assembly response to external sinusoidal and random vibration inputs, while accounting for damping and resonance of the strapdown inertial sensor assembly mechanical mounting within the INS chassis. Section 10.5 describes simplified analytical methods that can be used to model the damping/resonance dynamic response characteristics of the strapdown sensor assembly to system level linear/angular vibration inputs. Section 10.6 describes the structure of a simplified simulation program that can be used to evaluate vibration sensitive performance parameters based on the Section 10.3 - 10.5 results.

All the vectors in this chapter represent projections on the sensor assembly "body" B Frame coordinate axes as it is defined in Section 2.2. The B superscript notation on the vectors has been omitted in this chapter for simplicity.

10.1 RESPONSE TO DISCRETE SINUSOIDAL SENSOR VIBRATION INPUTS

In this section we analyze the effect of sensor assembly linear and angular vibration based on sinusoidal motion specified in the sensor assembly B Frame. The section is divided into five major subsections covering attitude motion response, velocity motion response, position motion response, vibration induced sensor error effects, and a summary of results obtained. For the attitude, velocity and position response subsections, discussions are provided on the true response, the response produced by the strapdown attitude/velocity/position integration algorithms, and the error induced by approximations in the integration algorithms.

10.1.1 ATTITUDE MOTION RESPONSE UNDER ANGULAR VIBRATION

To simplify the analysis to follow, we will restrict the attitude motion discussion to angular response generated by angular vibration around the B Frame X and Y axes. The results can then be easily extended by designated axis permutation to account for similar effects under Y/Z and Z/X angular vibrations.

We define the sinusoidal angular vibration profile to be analyzed by the following general forms:

$$\underline{\theta}(t) = \underline{u}_X \, \theta_{0_x} \sin\left(\Omega_x \, t - \varphi_{\theta_x}\right) + \underline{u}_Y \, \theta_{0_y} \sin\left(\Omega_y \, t - \varphi_{\theta_y}\right) \tag{10.1.1-1}$$

$$\underline{\omega}_{IB}(t) = \underline{u}_X \, \theta_{0_x} \, \Omega_x \cos\left(\Omega_x \, t - \varphi_{\theta_x}\right) + \underline{u}_Y \, \theta_{0_y} \, \Omega_y \cos\left(\Omega_y \, t - \varphi_{\theta_y}\right) \tag{10.1.1-2}$$

where

$\underline{\omega}_{IB}(t)$ = B Frame angular rate vector relative to inertial space that would be measured by the strapdown angular rate sensors.

$\underline{\theta}(t)$ = B Frame vibration "angle" vector which we define as the integrated B Frame angular rate. Note that $\underline{\omega}_{IB}(t)$ is the derivative of $\underline{\theta}(t)$. Since we will be addressing angular vibration effects that are by nature, small in amplitude, the integral of generalized Equation (3.3.5-14) shows that $\underline{\theta}(t)$ is approximately the rotation vector associated with the vibration motion, hence, represents an actual physical angle vector.

$\underline{u}_X, \underline{u}_Y$ = Unit vectors along the B Frame X, Y axes.

Ω_x, Ω_y = Frequency of the sinusoidal angular vibration around B Frame axes X and Y.

$\theta_{0_x}, \theta_{0_y}$ = Sinusoidal vibration "angle" vector amplitude around B Frame axes X and Y.

$\varphi_{\theta_x}, \varphi_{\theta_y}$ = Phase angle associated with each B Frame X, Y axis angular vibration.

10.1.1.1 ATTITUDE MOTION CHARACTERISTICS

Let us analyze the effect of the Equation (10.1.1-2) angular rate on B Frame attitude at time t relative to B Frame attitude at some arbitrary time t_0. To do this, it is convenient to define attitude in terms of a rotation vector described formally in Section 3.2.2, recognizing that any of the attitude parameters discussed in previous chapters (e.g. direction cosine matrix, attitude quaternion) can be analytically defined in terms of the rotation vector. To simplify the analysis, we recognize that attitude motion produced by angular vibration is, by nature, small in amplitude, hence, we can use Equation (7.1.1.1-10) with (7.1.1.1-9) as a model to write an approximate rotation vector rate equation:

$$\dot{\underline{\Phi}}(t) \approx \underline{\omega}_{IB}(t) + \frac{1}{2} \underline{\mathcal{A}}(t) \times \underline{\omega}_{IB}(t) \tag{10.1.1.1-1}$$

$$\underline{\mathcal{A}}(t) \equiv \int_{t_0}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \tag{10.1.1.1-2}$$

where

$\underline{\Phi}(t)$ = Rotation vector describing the B Frame attitude at time t relative to the B Frame attitude at some arbitrary prior initial time.

t_0 = Initial B Frame attitude reference time for $\underline{\Phi}(t)$ definition.

$\underline{\mathcal{A}}(t)$ = Integrated B Frame angular rate since time t_0.

The integral of (10.1.1.1-1) from time t_0 with (10.1.1.1-2) provides the rotation vector:

$$\underline{\Phi}(t) = \underline{\mathcal{A}}(t) + \int_{t_0}^{t} \frac{1}{2} \underline{\mathcal{A}}(\tau) \times \underline{\omega}_{IB}(\tau)\, d\tau \tag{10.1.1.1-3}$$

where

τ = Integration time parameter.

For the angular vibration motion described by Equations (10.1.1-1) - (10.1.1-2), Equation (10.1.1.1-2) and the integrand cross-product term in (10.1.1.1-3) are given by:

$$\begin{aligned} \underline{\mathcal{A}}(t) = \; & \underline{u}_x \, \theta_{0_x} \left[\sin\left(\Omega_x t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x t_0 - \varphi_{\theta_x}\right) \right] \\ & + \underline{u}_y \, \theta_{0_y} \left[\sin\left(\Omega_y t - \varphi_{\theta_y}\right) - \sin\left(\Omega_y t_0 - \varphi_{\theta_y}\right) \right] \end{aligned} \tag{10.1.1.1-4}$$

$$\begin{aligned} \underline{\mathcal{A}}(t) \times \underline{\omega}_{IB}(t) = \; & \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \Big\{ \left[\sin\left(\Omega_x t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x t_0 - \varphi_{\theta_x}\right) \right] \Omega_y \cos\left(\Omega_y t - \varphi_{\theta_y}\right) \\ & - \left[\sin\left(\Omega_y t - \varphi_{\theta_y}\right) - \sin\left(\Omega_y t_0 - \varphi_{\theta_y}\right) \right] \Omega_x \cos\left(\Omega_x t - \varphi_{\theta_x}\right) \Big\} \end{aligned} \tag{10.1.1.1-5}$$

or upon expansion for $\underline{\mathcal{A}}(t) \times \underline{\omega}_{IB}(t)$:

$$
\begin{aligned}
&\underline{\mathcal{A}}(t) \times \underline{\omega}_{IB}(t) \;= \\
&\qquad \underline{u}_Z\, \theta_{0_x}\, \theta_{0_y} \Big[-\,\Omega_y \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right) \cos\left(\Omega_y\, t - \varphi_{\theta_y}\right) + \Omega_x \,\sin\left(\Omega_y\, t_0 - \varphi_{\theta_y}\right) \cos\left(\Omega_x\, t - \varphi_{\theta_x}\right) \\
&\qquad\quad + \Omega_y \sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) \cos\left(\Omega_y\, t - \varphi_{\theta_y}\right) - \Omega_x \,\sin\left(\Omega_y\, t - \varphi_{\theta_y}\right) \cos\left(\Omega_x\, t - \varphi_{\theta_x}\right) \Big] \\
&= \underline{u}_Z\, \theta_{0_x}\, \theta_{0_y} \Bigg[-\,\Omega_y \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right) \cos\left(\Omega_y\, t - \varphi_{\theta_y}\right) + \Omega_x \,\sin\left(\Omega_y\, t_0 - \varphi_{\theta_y}\right) \cos\left(\Omega_x\, t - \varphi_{\theta_x}\right) \\
&\qquad + \frac{\Omega_y}{2} \Big\{ \sin\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] - \sin\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \Big\} \qquad (10.1.1.1\text{-}6) \\
&\qquad - \frac{\Omega_x}{2} \Big\{ \sin\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] + \sin\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \Big\} \Bigg] \\
&= \underline{u}_Z\, \theta_{0_x}\, \theta_{0_y} \Bigg\{ -\,\Omega_y \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right) \cos\left(\Omega_y\, t - \varphi_{\theta_y}\right) + \Omega_x \,\sin\left(\Omega_y\, t_0 - \varphi_{\theta_y}\right) \cos\left(\Omega_x\, t - \varphi_{\theta_x}\right) \\
&\qquad + \frac{\left(\Omega_y - \Omega_x\right)}{2} \sin\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] - \frac{\left(\Omega_y + \Omega_x\right)}{2} \sin\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \Bigg\}
\end{aligned}
$$

Substituting the (10.1.1.1-6) result into (10.1.1.1-3) then obtains for the integral:

$$
\begin{aligned}
&\int_{t_0}^{t} \frac{1}{2}\, \underline{\mathcal{A}}(\tau) \times \underline{\omega}_{IB}(\tau)\, d\tau \;=\; \underline{u}_Z\, \frac{1}{2}\, \theta_{0_x}\, \theta_{0_y} \Bigg[-\sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right) \left[\sin\left(\Omega_y\, t - \varphi_{\theta_y}\right) - \sin\left(\Omega_y\, t_0 - \varphi_{\theta_y}\right) \right] \\
&\qquad + \sin\left(\Omega_y\, t_0 - \varphi_{\theta_y}\right) \left[\sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right) \right] \\
&\qquad - \frac{\left(\Omega_y - \Omega_x\right)}{2\left(\Omega_y + \Omega_x\right)} \Big\{ \cos\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y + \Omega_x\right) t_0 - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \Big\} \\
&\qquad + \frac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)} \Big\{ \cos\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \Big\} \Bigg] \qquad (10.1.1.1\text{-}7) \\
&= \underline{u}_Z\, \frac{1}{2}\, \theta_{0_x}\, \theta_{0_y} \Bigg[-\sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right) \sin\left(\Omega_y\, t - \varphi_{\theta_y}\right) + \sin\left(\Omega_y\, t_o - \varphi_{\theta_y}\right) \sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) \\
&\qquad - \frac{\left(\Omega_y - \Omega_x\right)}{2\left(\Omega_y + \Omega_x\right)} \Big\{ \cos\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y + \Omega_x\right) t_0 - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \Big\} \\
&\qquad + \frac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)} \Big\{ \cos\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \Big\} \Bigg]
\end{aligned}
$$

The sine products in (10.1.1.1-7) can be expressed in an equivalent form as follows:

$$
\begin{aligned}
&- \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right) \sin\left(\Omega_y\, t - \varphi_{\theta_y}\right) + \sin\left(\Omega_y\, t_0 - \varphi_{\theta_y}\right) \sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) \\
&\quad = -\frac{1}{2} \cos\left[\Omega_y\, t - \Omega_x\, t_0 - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] + \frac{1}{2} \cos\left[\Omega_y\, t + \Omega_x\, t_0 - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \\
&\qquad + \frac{1}{2} \cos\left[\Omega_y\, t_0 - \Omega_x\, t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \frac{1}{2} \cos\left[\Omega_y\, t_0 + \Omega_x\, t - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \qquad (10.1.1.1\text{-}8) \\
&\quad = \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t - t_0\right)\right] \\
&\qquad - \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)\right]
\end{aligned}
$$

Substituting (10.1.1.1-4) and (10.1.1.1-7) with (10.1.1.1-8) into (10.1.1.1-3) then yields the equation for the rotation vector attitude response:

$$
\begin{aligned}
\underline{\Phi}(t) = \; & \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right)\right] \\
& + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega_y\, t - \varphi_{\theta_y}\right) - \sin\left(\Omega_y\, t_0 - \varphi_{\theta_y}\right)\right] \\
& + \underline{u}_z\, \frac{1}{2}\, \theta_{0_x}\, \theta_{0_y} \left\{ \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t - t_0\right)\right] \right. \\
& - \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)\right] \qquad (10.1.1.1\text{-}9) \\
& - \frac{\left(\Omega_y - \Omega_x\right)}{2\left(\Omega_y + \Omega_x\right)} \left(\cos\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y + \Omega_x\right) t_0 - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right]\right) \\
& \left. + \frac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)} \left(\cos\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right]\right) \right\}
\end{aligned}
$$

Equation (10.1.1.1-9) shows that for the hypothesized Equation (10.1.1-2) general sinusoidal X/Y angular rate vibration profile, the rotation vector attitude response is also sinusoidal at the same frequencies around B Frame axes X and Y. Around the B Frame Z axis, the response is the sum and the sum of products of sinusoids at the sum and difference frequencies $\left(\Omega_y + \Omega_x\right)$, $\left(\Omega_y - \Omega_x\right)$. Hence, the attitude response has no net rotation angle build-up, provided that the frequency difference $\left(\Omega_y - \Omega_x\right)$ is non-zero (i.e., $\Omega_x \neq \Omega_y$). For the case when Ω_x approaches Ω_y, the frequency difference cosine functions become very low in frequency, approaching a constant, and the associated multiplication term $\frac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)}$ becomes large. In the limit, when

Ω_x and Ω_y are equal, the product of $\dfrac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)}$ with the frequency difference cosines produces a linearly increasing value for the Z component of $\underline{\Phi}(t)$ as the following analysis shows.

For the case when Ω_x approaches Ω_y and in the limit, equals Ω_y, Equation (10.1.1.1-9) becomes:

For $\Omega_x \rightarrow \Omega_y = \Omega$:

$$\begin{aligned}
\underline{\Phi}(t) = \ & \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right)\right] \\
& + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_y}\right)\right] \\
& + \underline{u}_z \frac{1}{2}\, \theta_{0_x}\, \theta_{0_y} \Bigg\{ - \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \sin\left(\Omega\,(t - t_0)\right) \\
& + \frac{\Omega}{\left(\Omega_y - \Omega_x\right)} \left(\cos\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right]\right)\Bigg\}
\end{aligned} \tag{10.1.1.1-10}$$

where

Ω = Frequency for both B Frame X and Y axis angular vibrations.

The cosine terms in Equation (10.1.1.1-10) can be reduced further as $\Omega_x \rightarrow \Omega_y = \Omega$:

$$\begin{aligned}
& \cos\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \cos\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \\
& \quad = -2 \sin \frac{1}{2}\left[\left(\Omega_y - \Omega_x\right)\left(t + t_0\right) - 2\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] \sin \frac{1}{2}\left[\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)\right] \\
& \quad \approx \left(\Omega_y - \Omega_x\right)\left(t - t_0\right) \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)
\end{aligned} \tag{10.1.1.1-11}$$

Thus, with (10.1.1.1-11), when Ω_x and Ω_y are equal, the (10.1.1.1-10) rotation vector attitude history is as follows:

For $\Omega_x \rightarrow \Omega_y = \Omega$:

$$\begin{aligned}
\underline{\Phi}(t) = \ & \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right)\right] \\
& + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_y}\right)\right] \\
& + \underline{u}_z \frac{1}{2}\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left[\left(t - t_0\right) - \frac{\sin\left(\Omega\,(t - t_0)\right)}{\Omega}\right]
\end{aligned} \tag{10.1.1.1-12}$$

Comparing Equation (10.1.1.1-12) with general Equation (10.1.1.1-9), we see that the X and Y axis angular responses are sinusoidal and identical, but that for $\Omega_x \to \Omega_y = \Omega$ in (10.1.1.1-12), the Z axis response has been converted into an Ω frequency sinusoid plus an unbounded linear build-up with time $(t - t_0)$ function. After one vibration cycle (i.e., $t - t_0 > 2\,\pi / \Omega$) the linear $(t - t_0)$ function dominates the response. The (10.1.1.1-12) Z axis linear with time build-up term has been designated as the "coning" response. The term "coning" describes the conical surface generated by the B Frame Z axis under the Equation (10.1.1-1) and (10.1.1-2) angular motion when $\Omega_x = \Omega_y$. The linear with time coefficient in Equation (10.1.1.1-12) has been designated as the "coning rate" given by:

$$\underline{\dot{\Phi}}_{Con} = \underline{u}_Z \frac{1}{2} \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \qquad (10.1.1.1\text{-}13)$$

where

$\underline{\dot{\Phi}}_{Con}$ = Coning rate vector for B Frame X, Y axis angular vibration at the same frequency.

Note in Equation (10.1.1.1-13), that $\underline{\dot{\Phi}}_{Con}$ is proportional to the sine of the phase angle difference between the Equation (10.1.1-1) or (10.1.1-2) X and Y axis angular vibration components, with worst case coning rate occurring for a 90 degree phase separation. Note also that the coning rate amplitude is proportional to the product of the coning frequency with the angular vibration amplitudes around axes X and Y. Thus, coning rate increases linearly with the angular vibration frequency Ω (sometimes denoted as the "coning frequency") and as the square of the angular vibration amplitude. It is also interesting to note as can be demonstrated analytically, that under Equation (10.1.1-1) angular vibration (with $\Omega_x \to \Omega_y = \Omega$), the area traced out by the B Frame Z axis on the B Frame X, Y axis plane over one vibration cycle equals $\pi \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)$. The time rate that the area is swept is the previous expression multiplied by the vibration frequency (in Hz), which equals $\frac{\Omega}{2\,\pi} \times \pi \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)$ or $\frac{1}{2} \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)$, the coning rate magnitude in Equation (10.1.1.1-13).

10.1.1.2 ATTITUDE ALGORITHM RESPONSE

The software in a strapdown INS processes a digital integration algorithm to calculate B Frame attitude. For consistency with Section 10.1.1.1, we will assume that the attitude integration algorithm is a digital version of the continuous form Equations (10.1.1.1-2) and

(10.1.1.1-3) based on attitude in the form of a rotation vector. Thus, we assume the attitude algorithm is derived from the following equivalent version of (10.1.1.1-2) and (10.1.1.1-3):

$$\underline{\alpha}(t) \equiv \int_{t_{m-1}}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \underline{\alpha}_m = \underline{\alpha}(t_m) \qquad \underline{\mathcal{A}}_{m-1} = \sum_{1}^{m-1} \underline{\alpha}_i$$

$$\underline{\mathcal{A}}(t) = \underline{\mathcal{A}}_{m-1} + \underline{\alpha}(t)$$

$$\Delta\underline{\Phi}_m = \underline{\alpha}_m + \int_{t_{m-1}}^{t_m} \frac{1}{2}\, \underline{\mathcal{A}}(t) \times \underline{\omega}_{IB}(t)\, dt \qquad (10.1.1.2\text{-}1)$$

$$\underline{\Phi}_m = \sum_{1}^{m} \Delta\underline{\Phi}_i$$

where

m =Strapdown software attitude update algorithm cycle index. The m zero cycle corresponds to time t_0 in Equations (10.1.1.1-2) and (10.1.1.1-3).

i = Particular m cycle over the m cycle history.

$\underline{\alpha}(t)$ = Integrated B Frame angular rate from the last m cycle time t_{m-1} to time t.

$\underline{\alpha}_m$ = Integrated B Frame angular rate from t_{m-1} to time t_m.

$\underline{\Phi}_m$ = B Frame attitude at computer cycle m.

$\Delta\underline{\Phi}_m$ = Change in B Frame attitude over the m cycle time interval from t_{m-1} to t_m.

The integral term in the $\Delta\underline{\Phi}_m$ expression can be expanded if we substitute for $\underline{\mathcal{A}}(t)$ and apply the $\underline{\alpha}_m$ definition from (10.1.1.2-1):

$$\begin{aligned}\int_{t_{m-1}}^{t_m} \frac{1}{2}\, \underline{\mathcal{A}}(t) \times \underline{\omega}_{IB}(t)\, dt &= \int_{t_{m-1}}^{t_m} \frac{1}{2} \left(\underline{\mathcal{A}}_{m-1} + \underline{\alpha}(t)\right) \times \underline{\omega}_{IB}(t)\, dt \\ &= \frac{1}{2}\, \underline{\mathcal{A}}_{m-1} \times \underline{\alpha}_m + \int_{t_{m-1}}^{t_m} \frac{1}{2}\, \underline{\alpha}(t) \times \underline{\omega}_{IB}(t)\, dt = \frac{1}{2}\, \underline{\mathcal{A}}_{m-1} \times \underline{\alpha}_m + \underline{\beta}_m\end{aligned} \qquad (10.1.1.2\text{-}2)$$

with

$$\underline{\beta}_m \equiv \int_{t_{m-1}}^{t_m} \frac{1}{2}\, \underline{\alpha}(t) \times \underline{\omega}_{IB}(t)\, dt \qquad (10.1.1.2\text{-}3)$$

where

$\underline{\beta}_m$ = Coning contribution to $\Delta\underline{\Phi}_m$.

With (10.1.1.2-2) and (10.1.1.2-3), Equations (10.1.1.2-1) become:

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \underline{\alpha}_m = \underline{\alpha}(t_m) \qquad \underline{\mathcal{A}}_m = \sum_{1}^{m} \underline{\alpha}_i$$

$$\underline{\beta}_m = \int_{t_{m-1}}^{t_m} \frac{1}{2}\, \underline{\alpha}(t) \times \underline{\omega}_{IB}(t)\, dt \tag{10.1.1.2-4}$$

$$\underline{\Phi}_m = \underline{\mathcal{A}}_m + \sum_{1}^{m} \frac{1}{2}\, \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i + \sum_{1}^{m} \underline{\beta}_i$$

Equations (10.1.1.2-4) represent the equivalent digital forms of Equations (10.1.1.1-2) and (10.1.1.1-3), generating the identical solution at the m cycle times. The $\underline{\alpha}$, $\underline{\beta}$ terms in (10.1.1.2-4) should be recognized as the identical terms in Section 7.1.1.1 for strapdown attitude algorithm development. The following subsections analyze the response of Equations (10.1.1.2-4) under the hypothesized Section 10.1.1 sinusoidal angular vibrations for the exact implementation of (10.1.1.2-4) and for the approximate forms typified in the INS software.

10.1.1.2.1 Exact Attitude Algorithm Response

The (10.1.1.2-4) algorithm is exact, hence, its overall response at the m cycle times to general X, Y sinusoidal angular vibration is identical to (10.1.1.1-9) which was derived from (10.1.1.1-2) - (10.1.1.1-3). Similarly, the overall response for equal X, Y angular vibration frequencies is identical to the (10.1.1.1-12) - (10.1.1.1-13) results. It is instructive to analyze the individual contributions in (10.1.1.2-4) to the overall result. For this discussion, we will only address the case for which the X, Y angular vibration frequencies are equal (i.e., $\Omega_x = \Omega_y = \Omega$).

From Equation (10.1.1.1-4) (the exact equivalent to the (10.1.1.2-4) $\underline{\mathcal{A}}_m$ expression at t_m), we can write for $\underline{\mathcal{A}}_m$ under the (10.1.1-2) vibration exposure with $\Omega_x \rightarrow \Omega_y = \Omega$:

$$\begin{aligned} \underline{\mathcal{A}}_m = {} & \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_m - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right) \right] \\ & + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t_m - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_y}\right) \right] \end{aligned} \tag{10.1.1.2.1-1}$$

From the definition for $\underline{\alpha}_m$ in (10.1.1.2-4), we also write for the (10.1.1-2) vibration with $\Omega_x = \Omega_y = \Omega$:

$$\underline{\alpha}_m = \int_{t_{m-1}}^{t_m} \underline{\omega}_{IB}(t)\, dt = \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_m - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{m-1} - \varphi_{\theta_x}\right)\right] + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t_m - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{m-1} - \varphi_{\theta_y}\right)\right] \quad (10.1.1.2.1\text{-}2)$$

The combination of (10.1.1.2.1-1) and (10.1.1.2.1-2) for m = i then yields an expression for the cross-product term in the (10.1.1.2-4) $\underline{\Phi}_m$ equation:

$$\underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i = \quad (10.1.1.2.1\text{-}3)$$

$$\underline{u}_z\, \theta_{0_x}\, \theta_{0_y} \Big\{\left[\sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right)\right]\left[\sin\left(\Omega\, t_i - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_y}\right)\right]$$
$$- \left[\sin\left(\Omega\, t_{i-1} - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_y}\right)\right]\left[\sin\left(\Omega\, t_i - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right)\right]\Big\}$$

or with much manipulation:

$$\begin{aligned}
\underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i = \ & \underline{u}_z\, \theta_{0_x}\, \theta_{0_y} \Big\{\sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right) \sin\left(\Omega\, t_i - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right) \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_y}\right) \\
& - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right) \sin\left(\Omega\, t_i - \varphi_{\theta_y}\right) + \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right) \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_y}\right) \\
& - \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_y}\right) \sin\left(\Omega\, t_i - \varphi_{\theta_x}\right) + \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_y}\right) \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right) \\
& + \sin\left(\Omega\, t_0 - \varphi_{\theta_y}\right) \sin\left(\Omega\, t_i - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_y}\right) \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right)\Big\} \\
= \ & \underline{u}_z\, \theta_{0_x}\, \theta_{0_y} \Big\{\frac{1}{2} \cos\left[\Omega\left(t_i - t_{i-1}\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \frac{1}{2} \cos\left[\Omega\left(t_i + t_{i-1}\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \\
& - \frac{1}{2} \cos\left[\Omega\left(t_i - t_0\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] + \frac{1}{2} \cos\left[\Omega\left(t_i + t_0\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \qquad (10.1.1.2.1\text{-}4) \\
& + \frac{1}{2} \cos\left[\Omega\left(t_{i-1} - t_0\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \frac{1}{2} \cos\left[\Omega\left(t_{i-1} + t_0\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \\
& - \frac{1}{2} \cos\left[\Omega\left(t_i - t_{i-1}\right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] + \frac{1}{2} \cos\left[\Omega\left(t_i + t_{i-1}\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \\
& + \frac{1}{2} \cos\left[\Omega\left(t_i - t_0\right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \frac{1}{2} \cos\left[\Omega\left(t_i + t_0\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \\
& - \frac{1}{2} \cos\left[\Omega\left(t_{i-1} - t_0\right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] + \frac{1}{2} \cos\left[\Omega\left(t_{i-1} + t_0\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right]\Big\}
\end{aligned}$$

(Continued)

$$= \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \Big\{ \frac{1}{2} \cos\Big[\Omega\,(t_i - t_{i-1}) - \big(\varphi_{\theta_y} - \varphi_{\theta_x}\big)\Big] - \frac{1}{2} \cos\Big[\Omega\,(t_i - t_{i-1}) + \big(\varphi_{\theta_y} - \varphi_{\theta_x}\big)\Big]$$
$$- \frac{1}{2} \cos\Big[\Omega\,(t_i - t_0) - \big(\varphi_{\theta_y} - \varphi_{\theta_x}\big)\Big] + \frac{1}{2} \cos\Big[\Omega\,(t_i - t_0) + \big(\varphi_{\theta_y} - \varphi_{\theta_x}\big)\Big] \qquad (10.1.1.2.1\text{-}4)\ \text{(Continued)}$$
$$+ \frac{1}{2} \cos\Big[\Omega\,(t_{i-1} - t_0) - \big(\varphi_{\theta_y} - \varphi_{\theta_x}\big)\Big] - \frac{1}{2} \cos\Big[\Omega\,(t_{i-1} - t_0) + \big(\varphi_{\theta_y} - \varphi_{\theta_x}\big)\Big] \Big\}$$

$$= \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \Big\{ \frac{1}{2} \cos\Big[\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) - \Omega\,(t_i - t_{i-1})\Big] - \frac{1}{2} \cos\Big[\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) + \Omega\,(t_i - t_{i-1})\Big]$$
$$- \frac{1}{2} \cos\Big[\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) - \Omega\,(t_i - t_0)\Big] + \frac{1}{2} \cos\Big[\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) + \Omega\,(t_i - t_0)\Big]$$
$$+ \frac{1}{2} \cos\Big[\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) - \Omega\,(t_{i-1} - t_0)\Big] - \frac{1}{2} \cos\Big[\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) + \Omega\,(t_{i-1} - t_0)\Big] \Big\}$$

$$= \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \sin\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) \Big[\sin\,\Omega\,(t_i - t_{i-1}) - \sin\,\Omega\,(t_i - t_0) + \sin\,\Omega\,(t_{i-1} - t_0)\Big]$$

With (10.1.1.2.1-4), the summation term in the (10.1.1.2-4) $\underline{\Phi}_m$ expression becomes:

$$\sum_1^m \frac{1}{2}\, \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i = \underline{u}_z \frac{1}{2} \theta_{0_x} \, \theta_{0_y} \sin\big(\varphi_{\theta_y} - \varphi_{\theta_x}\big) \sum_1^m \Big[\sin\,\Omega\,(t_i - t_{i-1})$$
$$- \sin\,\Omega\,(t_i - t_0) + \sin\,\Omega\,(t_{i-1} - t_0)\Big] \qquad (10.1.1.2.1\text{-}5)$$

The (10.1.1.2.1-5) result can be further simplified using the following substitutions:

$$t_i - t_{i-1} = T_m \qquad t_i - t_0 = i\,T_m$$
$$t_{i-1} - t_0 = (i - 1)\,T_m \qquad m = \frac{t_m - t_0}{T_m} \qquad (10.1.1.2.1\text{-}6)$$

where

T_m = Time interval for the computer m cycle attitude update rate.

Substituting (10.1.1.2.1-6) into the (10.1.1.2.1-5) summation term gives:

$$\sum_{i=1}^m \Big[\sin\,\Omega\,T_m - \sin\,i\,\Omega\,T_m + \sin\,(i\text{-}1)\,\Omega\,T_m\Big]$$
$$= m \sin\Omega\,T_m - \sum_{i=1}^m \sin\,i\,\Omega\,T_m + \sum_{i=1}^m \sin\,(i\text{-}1)\,\Omega\,T_m \qquad (10.1.1.2.1\text{-}7)$$

(Continued)

$$= \; m \sin \Omega\, T_m - \sum_{i=1}^{m} \sin i\, \Omega\, T_m + \sum_{i=0}^{m-1} \sin i\, \Omega\, T_m \qquad \text{(10.1.1.2.1-7) (Continued)}$$

$$= \; m \sin \Omega\, T_m - \sin m\, \Omega\, T_m \;=\; \frac{(t_m - t_0)}{T_m} \sin \Omega\, T_m - \sin \Omega\,(t_m - t_0)$$

$$= \; \Omega \left(\frac{\sin \Omega\, T_m}{\Omega\, T_m} (t_m - t_0) - \frac{\sin \Omega\,(t_m - t_0)}{\Omega} \right)$$

With (10.1.1.2.1-7) substituted in (10.1.1.2.1-5), the summation term in the (10.1.1.2-4) $\underline{\Phi}_m$ expression finally becomes:

$$\sum_{1}^{m} \frac{1}{2} \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i \;=\; \underline{u}_Z \frac{1}{2} \Omega\, \theta_{0_x} \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left(\frac{\sin \Omega\, T_m}{\Omega\, T_m} (t_m - t_0) - \frac{\sin \Omega\,(t_m - t_0)}{\Omega} \right) \qquad \text{(10.1.1.2.1-8)}$$

An analytical equation for the (10.1.1.2-3) $\underline{\beta}_m$ coning term in the (10.1.1.2-4) $\underline{\Phi}_m$ expression is more easily obtained by direct extension of Equation (10.1.1.1-12), the solution to (10.1.1.1-3) under (10.1.1-2) vibration with $\Omega_x = \Omega_y = \Omega$. On review of the (10.1.1.1-12) derivation, it should be clear that the Z component of (10.1.1.1-12) is the integral term in (10.1.1.1-3), or with (10.1.1.1-2) and evaluation at t_m:

$$\underline{\mathcal{A}}(t) \;=\; \int_{t_0}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \text{(10.1.1.2.1-9)}$$

$$\int_{t_0}^{t_m} \frac{1}{2} \underline{\mathcal{A}}(t) \times \underline{\omega}_{IB}(t)\, dt \;=\; \underline{u}_Z \frac{1}{2} \Omega\, \theta_{0_x} \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left[(t_m - t_0) - \frac{\sin\left(\Omega\,(t_m - t_0)\right)}{\Omega} \right]$$

For comparison, the calculations for $\underline{\beta}_m$ in Equations (10.1.1.2-4) are repeated below:

$$\underline{\beta}_m \;=\; \int_{t_{m-1}}^{t_m} \frac{1}{2} \underline{\alpha}(t) \times \underline{\omega}_{IB}(t)\, dt \qquad \underline{\alpha}(t) \;=\; \int_{t_{m-1}}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \text{(10.1.1.2.1-10)}$$

Comparing Equations (10.1.1.2.1-9) and (10.1.1.2.1-10) we see that they are analytically equivalent; (10.1.1.2.1-9) can be converted to (10.1.1.2.1-10) through the following substitutions:

$$\underline{\mathcal{A}}(t) \;\rightarrow\; \underline{\alpha}(t) \qquad t_0 \;\rightarrow\; t_{m-1} \qquad t_m - t_{m-1} \;=\; T_m \tag{10.1.1.2.1-11}$$

for which the $\underline{\beta}_m$ analytical solution becomes:

$$\underline{\beta}_m \;=\; \underline{u}_z \, \frac{1}{2} \, \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) T_m \tag{10.1.1.2.1-12}$$

Thus, $\underline{\beta}_m$ is constant and the $\underline{\beta}_i$ summation term in the (10.1.1.2-4) $\underline{\Phi}_m$ expression is given by:

$$\begin{aligned} \sum_1^m \underline{\beta}_i \;&=\; m \, \underline{\beta}_m \;=\; \frac{\left(t_m - t_0\right)}{T_m} \, \underline{\beta}_m \\ &=\; \underline{u}_z \, \frac{1}{2} \, \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) \left(t_m - t_0\right) \end{aligned} \tag{10.1.1.2.1-13}$$

Because $\underline{\beta}_m$ is constant, we can also define a $\underline{\Phi}_m$ attitude build-up rate contribution associated with the $\underline{\beta}_i$ coning summation term in (10.1.1.2-4) as the $\left(t_m - t_0\right)$ coefficient in (10.1.1.2.1-13):

$$\dot{\underline{\beta}}_m \;=\; \underline{u}_z \, \frac{1}{2} \, \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) \tag{10.1.1.2.1-14}$$

where

$\dot{\underline{\beta}}_m$ = Constant rate of change in $\underline{\Phi}_m$ generated by the summing of $\underline{\beta}_i$'s in Equation (10.1.1.2-4).

As an exercise, we now form $\underline{\Phi}_m$ in (10.1.1.2-4) by combining (10.1.1.2.1-13), (10.1.1.2.1-8) and (10.1.1.2.1-1):

$$\begin{aligned} \underline{\Phi}_m \;=\; &\underline{u}_x \, \theta_{0_x} \left[\sin\left(\Omega \, t_m - \varphi_{\theta_x}\right) - \sin\left(\Omega \, t_0 - \varphi_{\theta_x}\right)\right] \\ &+ \underline{u}_y \, \theta_{0_y} \left[\sin\left(\Omega \, t_m - \varphi_{\theta_y}\right) - \sin\left(\Omega \, t_0 - \varphi_{\theta_y}\right)\right] \\ &+ \underline{u}_z \, \frac{1}{2} \, \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left((t_m - t_0) - \frac{\sin\left(\Omega \, (t_m - t_0)\right)}{\Omega} \right) \end{aligned} \tag{10.1.1.2.1-15}$$

The result is identical to (10.1.1.1-12) at the m cycle times (as it should be since (10.1.1.2-4) is an exact algorithm).

Let us now review the results of our findings. In Section 10.1.1.1 we noted that for time greater than one vibration cycle, the $\left(t_m - t_0\right)$ term in (10.1.1.2.1-15) dominates the $\underline{\Phi}_m$ Z axis response, having a linear with time "coning rate" build-up with slope:

$$\dot{\underline{\Phi}}_{Con} = \underline{u}_z \frac{1}{2} \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \tag{10.1.1.2.1-16}$$

Comparing $\dot{\underline{\beta}}_m$ in Equation (10.1.1.2.1-14) and $\dot{\underline{\Phi}}_{Con}$ in (10.1.1.2.1-16), we see that:

$$\dot{\underline{\beta}}_m = \left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) \dot{\underline{\Phi}}_{Con} \tag{10.1.1.2.1-17}$$

Thus, $\dot{\underline{\beta}}_m$ measures the $\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)$ portion of $\dot{\underline{\Phi}}_{Con}$. For large ΩT_m (i.e., large coning rate compared to the attitude update frequency $1 / T_m$), the $\frac{\sin \Omega T_m}{\Omega T_m}$ term in (10.1.1.2.1-17) goes to zero and $\dot{\underline{\beta}}_m$ becomes equal to the full coning rate $\dot{\underline{\Phi}}_{Con}$. For small ΩT_m, the $\frac{\sin \Omega T_m}{\Omega T_m}$ term goes to one, $\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)$ goes to zero, hence, $\dot{\underline{\beta}}_m$ goes to zero. Therefore, $\dot{\underline{\beta}}_m$ measures the high frequency portion of the total coning rate (i.e., high frequency compared to the attitude update frequency).

The total attitude is formed in the (10.1.1.2-4) $\underline{\Phi}_m$ equation by the summing of $\underline{\mathcal{A}}_m$ from (10.1.1.2.1-1) with $\sum_{1}^{m} \underline{\beta}_i$ from (10.1.1.2.1-13) and $\sum_{1}^{m} \frac{1}{2} \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i$ from (10.1.1.2.1-8). The coning contribution to $\underline{\Phi}_m$ is manifested in the $\dot{\underline{\Phi}}_{Con}$ build-up rate of Equation (10.1.1.2.1-16). From (10.1.1.2.1-1) we see that the $\underline{\mathcal{A}}_m$ term in (10.1.1.2-4) has no contribution to the (10.1.1.2.1-16) coning rate which is about $\underline{u}_z$. The contribution of $\sum_{1}^{m} \underline{\beta}_i$ to $\underline{\Phi}_m$ coning rate is $\dot{\underline{\beta}}_m$ in (10.1.1.2.1-17) which, as discussed previously, measures the $\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)$ high frequency portion of $\dot{\underline{\Phi}}_{Con}$. From (10.1.1.2.1-8) we see that the $\frac{\sin \Omega T_m}{\Omega T_m}$ portion of $\dot{\underline{\Phi}}_{Con}$ (i.e., the portion not measured by $\underline{\dot{\beta}}_m$) is contained in the $\sum_1^m \frac{1}{2} \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i$ term. Thus $\sum_1^m \frac{1}{2} \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i$ in (10.1.1.2-4) measures the low frequency portion of $\underline{\dot{\Phi}}_{Con}$.

A review of Section 7.1.1.1 reveals that $\underline{\beta}_m$ in (10.1.1.2-4) is the identical coning term measured by the high speed part of the two-speed attitude update algorithm (see Equations (7.1.1.1-12) - (7.1.1.1-13)). The low speed (m cycle) part of the attitude update algorithm measures the remaining $\underline{\mathcal{A}}_m + \sum_1^m \frac{1}{2} \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i$ term in (10.1.1.2-4). Thus, from (10.1.1.2.1-17), $\underline{\dot{\beta}}_m$ for the high speed algorithm measures $\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)$ of the total coning rate, while the remaining $\frac{\sin \Omega T_m}{\Omega T_m}$ coning rate portion is measured by the low speed part of the two-speed algorithm.

Equation (10.1.1.2.1-17) is based on an analytical integration for $\underline{\beta}_m$ in (10.1.1.2-4). In this sense, (10.1.1.2.1-17) can be considered as the solution for the “exact” coning algorithm for which exact denotes an infinitely fast computer capable of executing the continuous analytical integration operation. The low speed portion of the two-speed attitude updating algorithm was derived in Section 7.1.1.1 as an exact solution (assuming an exact input from the high speed portion). Thus, $\underline{\mathcal{A}}_m + \sum_1^m \frac{1}{2} \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i$ in (10.1.1.2-4) represents the low speed portion of both the exact version and the attitude updating algorithm version of Section 7.1.1.1 used in the INS computer.

10.1.1.2.2 <u>INS Attitude Algorithm Response And Error</u>

In the last section we analyzed the response of an exact attitude computation algorithm to hypothesized sinusoidal vibrations to discriminate between the total attitude solution and the portion of the attitude solution contributed by the exact $\underline{\beta}_m$ coning algorithm. We showed that under angular vibrations of identical frequency around the B Frame X and Y axes, the overall attitude rotated around the B Frame Z axis at a constant total “coning rate” $\underline{\dot{\Phi}}_{Con}$ described by Equation (10.1.1.2.1-16). In this section we will analyze the response of the attitude algorithm implemented in the strapdown INS which we model after the Equation (10.1.1.2-4) general form:

$$
\begin{aligned}
\underline{\mathcal{A}}_{Algo_m} &= \sum_1^m \underline{\alpha}_{Algo_i} \\
\underline{\Phi}_{Algo_m} &= \underline{\mathcal{A}}_{Algo_m} + \sum_1^m \frac{1}{2}\, \underline{\mathcal{A}}_{Algo_{i-1}} \times \underline{\alpha}_{Algo_i} + \sum_1^m \underline{\beta}_{Algo_i}
\end{aligned}
\qquad (10.1.1.2.2\text{-}1)
$$

where

$(\ )_{Algo}$ = Version of ( ) in Equations (10.1.1.2-4) implemented in the strapdown INS software.

Let us further assume the (10.1.1.2.2-1) INS software attitude algorithm is of the Section 7.1.1.1 two-speed type, with $\underline{\mathcal{A}}_{Algo_m} + \sum_1^m \frac{1}{2}\, \underline{\mathcal{A}}_{Algo_{i-1}} \times \underline{\alpha}_{Algo_i}$ implemented as an exact low-speed algorithm (as in Section 7.1.1.1), and the high speed portion implemented as in Equations (7.1.1.1.1-17) - (7.1.1.1.1-18):

$$
\begin{aligned}
&\Delta\underline{\alpha}_{Algo_l} = \int_{t_{l-1}}^{t_l} d\underline{\alpha} \qquad \text{Summation Of Integrated Angular Rate Output Increments From Angular Rate Sensors} \\
&\underline{\alpha}_{Algo_l} = \underline{\alpha}_{Algo_{l-1}} + \Delta\underline{\alpha}_{Algo_l} \\
&\underline{\alpha}_{Algo_m} = \underline{\alpha}_{Algo_l}(t_l = t_m) \qquad \underline{\alpha}_{Algo_l} = 0 \quad \text{At } t = t_{m-1}.
\end{aligned}
\qquad (10.1.1.2.2\text{-}2)
$$

$$
\begin{aligned}
&\Delta\underline{\beta}_{Algo_l} = \frac{1}{2}\left(\underline{\alpha}_{Algo_{l-1}} + \frac{1}{6}\, \Delta\underline{\alpha}_{Algo_{l-1}}\right) \times \Delta\underline{\alpha}_{Algo_l} \\
&\underline{\beta}_{Algo_l} = \underline{\beta}_{Algo_{l-1}} + \Delta\underline{\beta}_{Algo_l} \\
&\underline{\beta}_{Algo_m} = \underline{\beta}_{Algo_l}(t_l = t_m) \qquad \underline{\beta}_{Algo_l} = 0 \quad \text{At } t = t_{m-1}.
\end{aligned}
\qquad (10.1.1.2.2\text{-}3)
$$

If we now compare Equations (10.1.1.2.2-1) - (10.1.1.2.2-3) with the Equations (10.1.1.2-4) “exact” algorithm equivalent, it should be apparent that the algorithms are identical except for the $\underline{\beta}$ terms; i.e.:

$$
\Delta\underline{\alpha}_{Algo} = \Delta\underline{\alpha} \qquad \underline{\alpha}_{Algo} = \underline{\alpha} \qquad \underline{\mathcal{A}}_{Algo} = \underline{\mathcal{A}} \qquad (10.1.1.2.2\text{-}4)
$$

Using (10.1.1.2.2-4), we subtract the (10.1.1.2.2-1) and (10.1.1.2-4) $\underline{\Phi}$ total attitude expressions to obtain an equation for the INS attitude algorithm error:

$$\delta\underline{\Phi}_{Algo_m} \equiv \underline{\Phi}_{Algo_m} - \underline{\Phi}_m = \sum_1^m \underline{\beta}_{Algo_i} - \sum_1^m \underline{\beta}_i \tag{10.1.1.2.2-5}$$

where

$\delta\underline{\Phi}_{Algo_m}$ = Total INS attitude algorithm error at completion of attitude computation cycle m.

We also define:

$$\delta\underline{\beta}_{Algo_m} \equiv \underline{\beta}_{Algo_m} - \underline{\beta}_m \tag{10.1.1.2.2-6}$$

where

$\delta\underline{\beta}_{Algo_m}$ = INS coning algorithm error for attitude computation cycle m.

Let us evaluate the individual terms in (10.1.1.2.2-5) - (10.1.1.2.2-6) under our hypothesized (10.1.1-2) angular rate vibration exposure with $\Omega_x = \Omega_y = \Omega$. First, we find for the true integrated rate terms:

$$\begin{aligned}
\underline{\alpha}_l &\equiv \int_{t_{l=(m-1)s}}^{t_l} \underline{\omega}_{IB}(t)\, dt = \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_l - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{l=(m-1)s} - \varphi_{\theta_x}\right)\right] \\
&\qquad + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t_l - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{l=(m-1)s} - \varphi_{\theta_y}\right)\right] \\
\Delta\underline{\alpha}_l &\equiv \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}(t)\, dt = \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_l - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{l-1} - \varphi_{\theta_x}\right)\right] \\
&\qquad + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t_l - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{l-1} - \varphi_{\theta_y}\right)\right]
\end{aligned} \tag{10.1.1.2.2-7}$$

where

s = Number of *l* cycles in an m cycle.

We then find an analytical expression for $\underline{\beta}_{Algo_m}$ in (10.1.1.2.2-3) which we first rewrite using (10.1.1.2.2-4):

$$\underline{\beta}_{Algo_m} = \sum_{(m-1)s+1}^{ms} \frac{1}{2}\, \underline{\alpha}_{\,l-1} \times \Delta\underline{\alpha}_{\,l} + \sum_{(m-1)s+1}^{ms} \frac{1}{12}\, \Delta\underline{\alpha}_{\,l-1} \times \Delta\underline{\alpha}_{\,l} \tag{10.1.1.2.2-8}$$

Following the lengthy procedure that led to $\sum_1^m \frac{1}{2}\, \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i$ in Equation (10.1.1.2.1-8) from (10.1.1.2.1-1) - (10.1.1.2.1-2), Equations (10.1.1.2.2-7) can be combined to derive an analytical expression for the $\sum_{(m-1)s+1}^{ms} \frac{1}{2} \underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l$ term in (10.1.1.2.2-8). Alternatively, $\sum_{(m-1)s+1}^{ms} \frac{1}{2} \underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l$ can be determined by inspection if we compare (10.1.1.2.2-7) with Equations (10.1.1.2.1-1) - (10.1.1.2.1-2), and note that they are of identical form. This observation allows us to quickly write $\sum_{(m-1)s+1}^{ms} \frac{1}{2} \underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l$ from the $\sum_{1}^{m} \frac{1}{2} \underline{\mathcal{A}}_{i-1} \times \underline{\alpha}_i$ result by translating the (10.1.1.2.1-8) terms:

$$\underline{\mathcal{A}}_{i-1} \rightarrow \underline{\alpha}_{l-1} \qquad \underline{\alpha}_i \rightarrow \Delta\underline{\alpha}_l \qquad T_m \rightarrow T_l \qquad t_m \rightarrow t_{l=ms} \qquad t_0 \rightarrow t_{l=(m-1)s}$$

$$\left(t_m - t_0\right) \rightarrow \left(t_{l=ms} - t_{l=(m-1)s}\right) = s\ T_l = T_m \qquad (i = 1 \text{ to } m) \rightarrow \left[\, l = (m-1)s+1 \text{ to } ms \,\right]$$

Then we quickly find:

$$\sum_{(m-1)s+1}^{ms} \frac{1}{2} \underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l = \underline{u}_z \frac{1}{2} \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left(\frac{\sin\, \Omega\, T_l}{\Omega\, T_l} - \frac{\sin\, \Omega\, T_m}{\Omega\, T_m} \right) T_m \tag{10.1.1.2.2-9}$$

The $\sum_{(m-1)s+1}^{ms} \frac{1}{12} \Delta\underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l$ term in (10.1.1.2.2-8) is unique to the particular Section 7.1.1.1.1 second order coning algorithm. A more general higher order form of (10.1.1.2.2-8) for coning algorithms based on l cycle integrated angular rate measurements is:

$$\Delta\underline{\alpha}_j \equiv \int_{t_{j-1}}^{t_j} d\underline{\alpha} \qquad \Delta\underline{\alpha}_k \equiv \int_{t_{k-1}}^{t_k} d\underline{\alpha}$$
$$\underline{\beta}_{Algo_m} = \sum_{(m-1)s+1}^{ms} \frac{1}{2} \underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l + \sum_{l=(m-1)s+1}^{ms} \sum_{j,\,k} C_{j,k}\, \Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k \tag{10.1.1.2.2-10}$$

where

- j, k = High speed computer cycle indices at the l cycle rate that are displaced from the l cycle by integers.
- $C_{j,k}$ = Coefficient for the j, k product.

For the Section 7.1.1.1.1 second order coning algorithm represented by (10.1.1.2.2-8), the values for j, k and $C_{j,k}$ are $j = l\text{-}1$, $k = l$ and $C_{j,k} = 1/12$. Variations on (10.1.1.2.2-10) are also possible (e.g., References 11 and 12) in which the j, k computer cycle rate is faster than the *l* cycle rate by an integer multiple with the j, k summation performed once each *l* cycle. This latter approach requires the angular rate sensor $\Delta\underline{\alpha}$ summers to be sampled at the faster j, k cycle rate. Returning to our original j, k definition (i.e., at the *l* cycle rate), let us now develop a general expression for the summation of one of the j, k product terms in (10.1.1.2.2-10).

From the (10.1.1.2.2-10) definitions, we first write as in (10.1.1.2.2-7):

$$\begin{aligned}
\Delta\underline{\alpha}_j &= \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_j - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_x}\right)\right] \\
&\quad + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t_j - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_y}\right)\right] \\
\Delta\underline{\alpha}_k &= \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_k - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{\theta_x}\right)\right] \\
&\quad + \underline{u}_y\, \theta_{0_y} \left[\sin\left(\Omega\, t_k - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{\theta_y}\right)\right]
\end{aligned} \qquad (10.1.1.2.2\text{-}11)$$

Combining Equations (10.1.1.2.2-11) into the (10.1.1.2.2-10) inner summation cross-product gives:

$$\begin{aligned}
&\Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k = \qquad (10.1.1.2.2\text{-}12) \\
&\underline{u}_z\, \theta_{0_x}\, \theta_{0_y} \Big\{\left[\sin\left(\Omega\, t_j - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_x}\right)\right]\left[\sin\left(\Omega\, t_k - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{\theta_y}\right)\right] \\
&\quad - \left[\sin\left(\Omega\, t_j - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_y}\right)\right]\left[\sin\left(\Omega\, t_k - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{\theta_x}\right)\right]\Big\}
\end{aligned}$$

or upon expansion:

$$\begin{aligned}
\Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k &= \underline{u}_z\, \theta_{0_x}\, \theta_{0_y} \Big\{\sin\left(\Omega\, t_j - \varphi_{\theta_x}\right)\sin\left(\Omega\, t_k - \varphi_{\theta_y}\right) - \sin\left(\Omega\, t_j - \varphi_{\theta_x}\right)\sin\left(\Omega\, t_{k-1} - \varphi_{\theta_y}\right) \\
&\quad - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_x}\right)\sin\left(\Omega\, t_k - \varphi_{\theta_y}\right) + \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_x}\right)\sin\left(\Omega\, t_{k-1} - \varphi_{\theta_y}\right) \\
&\quad - \sin\left(\Omega\, t_j - \varphi_{\theta_y}\right)\sin\left(\Omega\, t_k - \varphi_{\theta_x}\right) + \sin\left(\Omega\, t_j - \varphi_{\theta_y}\right)\sin\left(\Omega\, t_{k-1} - \varphi_{\theta_x}\right) \qquad (10.1.1.2.2\text{-}13) \\
&\quad + \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_y}\right)\sin\left(\Omega\, t_k - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_y}\right)\sin\left(\Omega\, t_{k-1} - \varphi_{\theta_x}\right)\Big\} \\
&= \underline{u}_z\, \theta_{0_x}\, \theta_{0_y} \Big\{\frac{1}{2}\cos\left[\Omega\left(t_k - t_j\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] - \frac{1}{2}\cos\left[\Omega\left(t_k + t_j\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \\
&\quad - \frac{1}{2}\cos\left[\Omega\left(t_{k-1} - t_j\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] + \frac{1}{2}\cos\left[\Omega\left(t_{k-1} + t_j\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right] \\
&\quad - \frac{1}{2}\cos\left[\Omega\left(t_k - t_{j-1}\right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\right] + \frac{1}{2}\cos\left[\Omega\left(t_k + t_{j-1}\right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x}\right)\right]
\end{aligned}$$

(Continued)

$$
\begin{aligned}
& + \frac{1}{2} \cos \left[\Omega \left(t_{k-1} - t_{j-1} \right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] - \frac{1}{2} \cos \left[\Omega \left(t_{k-1} + t_{j-1} \right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x} \right) \right] \\
& - \frac{1}{2} \cos \left[\Omega \left(t_k - t_j \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] + \frac{1}{2} \cos \left[\Omega \left(t_k + t_j \right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x} \right) \right] \\
& + \frac{1}{2} \cos \left[\Omega \left(t_{k-1} - t_j \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] - \frac{1}{2} \cos \left[\Omega \left(t_{k-1} + t_j \right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x} \right) \right] \\
& + \frac{1}{2} \cos \left[\Omega \left(t_k - t_{j-1} \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] - \frac{1}{2} \cos \left[\Omega \left(t_k + t_{j-1} \right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x} \right) \right] \\
& \left. - \frac{1}{2} \cos \left[\Omega \left(t_{k-1} - t_{j-1} \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] + \frac{1}{2} \cos \left[\Omega \left(t_{k-1} + t_{j-1} \right) - \left(\varphi_{\theta_y} + \varphi_{\theta_x} \right) \right] \right\} \\
= \ & \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \left\{ \frac{1}{2} \cos \left[\Omega \left(t_k - t_j \right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] - \frac{1}{2} \cos \left[\Omega \left(t_k - t_j \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] \right. \\
& - \frac{1}{2} \cos \left[\Omega \left(t_{k-1} - t_j \right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] + \frac{1}{2} \cos \left[\Omega \left(t_{k-1} - t_j \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] \\
& - \frac{1}{2} \cos \left[\Omega \left(t_k - t_{j-1} \right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] + \frac{1}{2} \cos \left[\Omega \left(t_k - t_{j-1} \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] \\
& \left. + \frac{1}{2} \cos \left[\Omega \left(t_{k-1} - t_{j-1} \right) - \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] - \frac{1}{2} \cos \left[\Omega \left(t_{k-1} - t_{j-1} \right) + \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \right] \right\} \\
= \ & \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \sin \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \left\{ \sin \Omega \left(t_k - t_j \right) - \sin \Omega \left(t_{k-1} - t_j \right) - \sin \Omega \left(t_k - t_{j-1} \right) + \sin \Omega \left(t_{k-1} - t_{j-1} \right) \right\}
\end{aligned}
\qquad \text{(10.1.1.2.2-13) (Continued)}
$$

The Equation (10.1.1.2.2-13) result can be simplified if we introduce:

$$
p_{jk} \equiv \frac{t_k - t_j}{T_l} \qquad \text{(10.1.1.2.2-14)}
$$

where

p_{jk} = Integer representing the number of l cycles from t_j to t_k.

for which

$$
\begin{aligned}
& t_k - t_j = p_{jk} \, T_l \qquad && t_{k-1} - t_j = \left(p_{jk} - 1 \right) T_l \\
& t_k - t_{j-1} = \left(p_{jk} + 1 \right) T_l \qquad && t_{k-1} - t_{j-1} = p_{jk} \, T_l
\end{aligned}
\qquad \text{(10.1.1.2.2-15)}
$$

Using (10.1.1.2.2-15), the Equation (10.1.1.2.2-13) result reduces to:

$$
\begin{aligned}
\Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k = \ & \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \sin \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \left[\sin p_{jk} \, \Omega \, T_l - \sin \left(p_{jk} - 1 \right) \Omega \, T_l \right. \\
& \left. - \sin \left(p_{jk} + 1 \right) \Omega \, T_l + \sin p_{jk} \, \Omega \, T_l \right] \\
= \ & \underline{u}_z \, \theta_{0_x} \, \theta_{0_y} \sin \left(\varphi_{\theta_y} - \varphi_{\theta_x} \right) \left[2 \sin p_{jk} \, \Omega \, T_l - \sin \left(p_{jk} - 1 \right) \Omega \, T_l - \sin \left(p_{jk} + 1 \right) \Omega \, T_l \right]
\end{aligned}
\qquad \text{(10.1.1.2.2-16)}
$$

Equation (10.1.1.2.2-16) expresses the well known curious result (among coning algorithm designers) that for the Equation (10.1.1-2) angular rate with $\Omega_x = \Omega_y = \Omega$, the cross-product between integrated angular rate increments separated by any integer number of angular rate integration cycles, is constant. This then allows the (10.1.1.2.2-10) summing operation over l to be easily evaluated as:

$$\sum_{l=(m-1)s+1}^{ms} \Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k = \tag{10.1.1.2.2-17}$$
$$\underline{u}_Z\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left(2 \sin p_{jk}\, \Omega\, T_l - \sin\left(p_{jk} - 1\right)\Omega\, T_l - \sin\left(p_{jk} + 1\right)\Omega\, T_l\right) s$$

The s term in (10.1.1.2.2-17) is the number of l cycles in an m cycle, hence:

$$s = \frac{T_m}{T_l} \tag{10.1.1.2.2-18}$$

with which (10.1.1.2.2-17) assumes the form:

$$\sum_{l=(m-1)s+1}^{ms} \Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k = \tag{10.1.1.2.2-19}$$
$$\underline{u}_Z\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left(2\, \frac{\sin p_{jk}\, \Omega\, T_l}{\Omega\, T_l} - \frac{\sin\left(p_{jk} - 1\right)\Omega\, T_l}{\Omega\, T_l} - \frac{\sin\left(p_{jk} + 1\right)\Omega\, T_l}{\Omega\, T_l}\right) T_m$$

For the particular Section 7.1.1.1.1 coning algorithm represented by (10.1.1.2.2-8), $p_{jk} = 1$ and $C_{j,k} = 1/12$ so:

$$\begin{aligned} &\sum_{l=(m-1)s+1}^{ms} \frac{1}{12}\, \Delta\underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_l = \\ &\underline{u}_Z\, \frac{1}{12}\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left(2\, \frac{\sin\, \Omega\, T_l}{\Omega\, T_l} - \frac{\sin 2\, \Omega\, T_l}{\Omega\, T_l}\right) T_m \end{aligned} \tag{10.1.1.2.2-20}$$

The bracketed term in (10.1.1.2.2-20) simplifies as follows:

$$\begin{aligned} 2\, \frac{\sin\, \Omega\, T_l}{\Omega\, T_l} - \frac{\sin 2\, \Omega\, T_l}{\Omega\, T_l} &= 2\, \frac{\sin\, \Omega\, T_l}{\Omega\, T_l} - \frac{2 \sin\, \Omega\, T_l \cos\, \Omega\, T_l}{\Omega\, T_l} \\ &= 2\, \frac{\sin\, \Omega\, T_l}{\Omega\, T_l}\left(1 - \cos\, \Omega\, T_l\right) \end{aligned} \tag{10.1.1.2.2-21}$$

hence,

$$\sum_{l=(m-1)s+1}^{ms} \frac{1}{12}\, \Delta\underline{\alpha}_{l-1} \times \Delta\underline{\alpha}_{l} =$$
$$\underline{u}_Z \frac{1}{6}\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \frac{\sin \Omega\, T_l}{\Omega\, T_l} \left(1 - \cos \Omega\, T_l\right) T_m \tag{10.1.1.2.2-22}$$

Combining (10.1.1.2.2-9) with (10.1.1.2.2-22) in (10.1.1.2.2-8) and factoring 1/2 to the left, we obtain for the INS software coning algorithm solution $\underline{\beta}_{\mathrm{Algo}_m}$:

$$\begin{aligned}
\underline{\beta}_{\mathrm{Algo}_m} &= \underline{u}_Z \frac{1}{2}\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left[\left(\frac{\sin \Omega\, T_l}{\Omega\, T_l} - \frac{\sin \Omega\, T_m}{\Omega\, T_m} \right) \right. \\
&\qquad \left. + \frac{1}{3} \left(\frac{\sin \Omega\, T_l}{\Omega\, T_l} \left(1 - \cos \Omega\, T_l\right) \right) \right] T_m \qquad (10.1.1.2.2\text{-}23) \\
&= \underline{u}_Z \frac{1}{2}\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left\{ \left[1 + \frac{1}{3} \left(1 - \cos \Omega\, T_l\right) \right] \left(\frac{\sin \Omega\, T_l}{\Omega\, T_l} \right) - \frac{\sin \Omega\, T_m}{\Omega\, T_m} \right\} T_m
\end{aligned}$$

Since $\underline{\beta}_{\mathrm{Algo}_m}$ is constant in (10.1.1.2.2-23) and independent of m, its sum in (10.1.1.2.2-5) is easily evaluated as m times $\underline{\beta}_{\mathrm{Algo}_m}$. We identify the product of m with T_m as $t_m - t_0$ and have:

$$\begin{aligned}
\sum_{1}^{m} \underline{\beta}_{\mathrm{Algo}_i} &= \underline{u}_Z \frac{1}{2}\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left\{ \left[1 + \frac{1}{3} \left(1 - \cos \Omega\, T_l\right) \right] \left(\frac{\sin \Omega\, T_l}{\Omega\, T_l} \right) \right. \\
&\qquad \left. - \frac{\sin \Omega\, T_m}{\Omega\, T_m} \right\} (t_m - t_0) \qquad (10.1.1.2.2\text{-}24)
\end{aligned}$$

Finally, we substitute (10.1.1.2.2-24) and $\sum_{1}^{m} \underline{\beta}_i$ from (10.1.1.2.1-13) in (10.1.1.2.2-5) to obtain for the total INS attitude algorithm attitude error at computer cycle m:

$$\begin{aligned}
\delta\underline{\Phi}_{\mathrm{Algo}_m} &= \underline{u}_Z \frac{1}{2}\, \Omega\, \theta_{0_x}\, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right) \left\{ \left[1 \right. \right. \\
&\qquad \left. \left. + \frac{1}{3} \left(1 - \cos \Omega\, T_l\right) \right] \left(\frac{\sin \Omega\, T_l}{\Omega\, T_l} \right) - 1 \right\} (t_m - t_o)
\end{aligned} \tag{10.1.1.2.2-25}$$

We also substitute (10.1.1.2.2-23) for $\underline{\beta}_{\mathrm{Algo}_m}$ with (10.1.1.2.1-12) for $\underline{\beta}_m$ into (10.1.1.2.2-6) which gives the INS coning algorithm error for computer cycle m:

$$\delta\underline{\beta}_{Algo_m} = \underline{u}_Z \frac{1}{2}\,\Omega\,\theta_{0_x}\,\theta_{0_y}\,\sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos\Omega\,T_l\right)\right]\left(\frac{\sin\Omega\,T_l}{\Omega\,T_l}\right) - 1\right\} T_m \qquad (10.1.1.2.2\text{-}26)$$

Thus, we see from (10.1.1.2.2-25) and (10.1.1.2.2-26) that the errors in the INS total attitude solution and coning algorithm are proportional to the associated evaluation time interval, with the proportionality coefficient independent of time. As in Section 10.1.1.1, Equation (10.1.1.1-13) and Section 10.1.1.2.1, Equation (10.1.1.2.1-14), it is meaningful to identify the coefficients in (10.1.1.2.2-23), (10.1.1.2.2-25) and (10.1.1.2.2-26) as the coning algorithm rate, the attitude algorithm rate error and the coning algorithm rate error given by:

$$\dot{\underline{\beta}}_{Algo_m} = \underline{u}_Z \frac{1}{2}\,\Omega\,\theta_{0_x}\,\theta_{0_y}\,\sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos\Omega\,T_l\right)\right]\left(\frac{\sin\Omega\,T_l}{\Omega\,T_l}\right) - \frac{\sin\Omega\,T_m}{\Omega\,T_m}\right\} \qquad (10.1.1.2.2\text{-}27)$$

$$\delta\dot{\underline{\Phi}}_{Algo_m} = \delta\dot{\underline{\beta}}_{Algo_m} = \underline{u}_Z \frac{1}{2}\,\Omega\,\theta_{0_x}\,\theta_{0_y}\,\sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos\Omega\,T_l\right)\right]\left(\frac{\sin\Omega\,T_l}{\Omega\,T_l}\right) - 1\right\} \qquad (10.1.1.2.2\text{-}28)$$

where

$\dot{\underline{\beta}}_{Algo_m}$ = Constant rate of change in the INS total attitude algorithm solution $\underline{\Phi}_{Algo_m}$ generated by the summing of $\underline{\beta}_{Algo_i}$'s in Equation (10.1.1.2.2-1).

$\delta\dot{\underline{\Phi}}_{Algo_m}$, $\delta\dot{\underline{\beta}}_{Algo_m}$ = Error build-up rate associated with the INS software total attitude algorithm and the coning contribution to the total attitude algorithm.

10.1.2 VELOCITY RESPONSE UNDER COMBINED ANGULAR AND LINEAR VIBRATION

To simplify the analysis to follow, we will restrict the discussion to velocity generated by angular vibration around B Frame axis X coupled with linear vibration along B Frame axis Y. The results can then be easily extended by designated axis permutation to account for similar effect under Y/Z, Z/X and Y/X, Z/Y, X/Z angular/linear vibrations.

We define the sinusoidal angular/linear vibration profile being analyzed by the following general forms:

$$\underline{\theta}(t) = \underline{u}_x \, \theta_{0_x} \sin\left(\Omega_x \, t - \varphi_{\theta_x}\right) \tag{10.1.2-1}$$

$$\underline{\omega}_{IB}(t) = \underline{u}_x \, \theta_{0_x} \, \Omega_x \cos\left(\Omega_x \, t - \varphi_{\theta_x}\right) \tag{10.1.2-2}$$

$$\underline{a}_{SF}(t) = \underline{u}_y \, a_{SF0_y} \sin\left(\Omega_y \, t - \varphi_{a_{SF_y}}\right) \tag{10.1.2-3}$$

where

$\underline{\omega}_{IB}(t)$ = B Frame angular rate vector relative to inertial space that would be measured by the strapdown angular rate sensors.

$\underline{\theta}(t)$ = B Frame vibration "angle" vector which we define as the integrated B Frame angular rate. Note that $\underline{\omega}_{IB}(t)$ is the derivative of $\underline{\theta}(t)$. Since we will be addressing angular vibration effects that are by nature, small in amplitude, the integral of generalized Equation (3.3.5-14) shows that $\underline{\theta}(t)$ is approximately the rotation vector associated with the vibration motion, hence, represents an actual physical angle vector.

$\underline{a}_{SF}(t)$ = B Frame specific force acceleration vector that would be measured by the strapdown accelerometers.

$\underline{u}_x, \underline{u}_y$ = Unit vectors along the B Frame X, Y axes.

Ω_x, Ω_y = Frequency of the sinusoidal angular/linear vibrations around B Frame axis X and along B Frame axis Y.

θ_{0_x} = Sinusoidal vibration "angle" vector amplitude around B Frame axis X.

a_{SF0_y} = Sinusoidal vibration amplitude of the B Frame Y axis specific force acceleration vibration.

$\varphi_{\theta_x}, \varphi_{a_{SF_y}}$ = Phase angles associated with the B Frame X, Y axis angular/linear vibrations.

The velocity response analyses to follow almost directly parallel the analyses of Section 10.1.1 for attitude response under vibration. To make the parallels clearer, it is advantageous to use the following equivalent form of (10.1.2-3) for the acceleration vibration:

$$\underline{a}_{SF}(t) = \underline{u}_y \, a_{SF0_y} \cos\left(\Omega_y \, t - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) \tag{10.1.2-4}$$

10.1.2.1 VELOCITY MOTION CHARACTERISTICS

Let us analyze the effect of Equation (10.1.2-2) angular rate and Equation (10.1.2-4) linear acceleration on velocity motion at time t relative to some arbitrary time t_0. To do this, it is convenient to define velocity as the integral of specific force in a non-rotating coordinate frame. The non-rotating frame we select is the B Frame at time t_0. To simplify the analysis, we recognize that attitude motion produced by angular vibration is, by nature, small in amplitude, hence, we can use Equation (7.2.2.2-5) - (7.2.2.2-6) with (7.2.2.2-21) as a model to write an approximate equation for the velocity rate in the time t_0 oriented B Frame:

$$\dot{\underline{v}}_{SF}(t) \; = \; \underline{a}_{SF}(t) + \frac{d}{dt}\frac{1}{2}\left(\underline{\mathcal{A}}(t) \times \underline{\mathcal{V}}(t)\right) + \frac{1}{2}\left(\underline{\mathcal{A}}(t) \times \underline{a}_{SF}(t) + \underline{\mathcal{V}}(t) \times \underline{\omega}_{IB}(t)\right) \qquad (10.1.2.1\text{-}1)$$

with

$$\underline{\mathcal{A}}(t) \; \equiv \; \int_{t_0}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \underline{\mathcal{V}}(t) \; \equiv \; \int_{t_0}^{t} \underline{a}_{SF}(\tau)\, d\tau \qquad (10.1.2.1\text{-}2)$$

where

t_0 = Initial time for B Frame attitude reference and velocity definition.

$\underline{v}_{SF}(t)$ = Velocity at time t in the time t_0 oriented B Frame due to (10.1.2-2) and (10.1.2-4) angular/linear vibration since time t_0.

$\underline{\mathcal{A}}(t)$, $\underline{\mathcal{V}}(t)$ = Integrated B Frame angular rate and specific force acceleration since time t_0.

The integral of (10.1.2.1-1) from time t_0 with the (10.1.2.1-2) definitions, provides an expression for the velocity in the t_0 B Frame for analytical development:

$$\underline{v}_{SF}(t) \; = \; \underline{\mathcal{V}}(t) + \frac{1}{2}\underline{\mathcal{A}}(t) \times \underline{\mathcal{V}}(t) + \int_{t_0}^{t} \frac{1}{2}\left(\underline{\mathcal{A}}(\tau) \times \underline{a}_{SF}(\tau) + \underline{\mathcal{V}}(\tau) \times \underline{\omega}_{IB}(\tau)\right) d\tau \qquad (10.1.2.1\text{-}3)$$

We now substitute Equations (10.1.2-2) and (10.1.2-4) for the sinusoidal vibration components into Equation (10.1.2.1-3) with (10.1.2.1-2) and expand. For the (10.1.2.1-2) terms we find:

$$\begin{aligned} \underline{\mathcal{A}}(t) \; &= \; \underline{u}_x\, \theta_{0_x}\left[\sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right)\right] \\ \underline{\mathcal{V}}(t) \; &= \; \underline{u}_y\, a_{SF0_y}\frac{1}{\Omega_y}\left[\sin\left(\Omega_y\, t - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega_y\, t_0 - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right] \end{aligned} \qquad (10.1.2.1\text{-}4)$$

With (10.1.2.1-4), the $\frac{1}{2}\underline{\mathcal{A}}(t) \times \underline{\mathcal{V}}(t)$ term in (10.1.2.1-3) is:

$$\frac{1}{2}\underline{\mathcal{A}}(t) \times \underline{\mathcal{V}}(t) = \underline{u}_Z \frac{1}{2}\theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega_y}\left[\sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right)\right]\left[\cos\left(\Omega_y\, t_0 - \varphi_{a_{SF_y}}\right) - \cos\left(\Omega_y\, t - \varphi_{a_{SF_y}}\right)\right] \tag{10.1.2.1-5}$$

With (10.1.2.1-4), (10.1.2-2) and (10.1.2-4), the $\underline{\mathcal{A}}(t) \times \underline{a}_{SF}(t) + \mathcal{V}(t) \times \underline{\omega}_{IB}(t)$ term in the (10.1.2.1-3) integrand is:

$$\begin{aligned}
&\underline{\mathcal{A}}(t) \times \underline{a}_{SF}(t) + \mathcal{V}(t) \times \underline{\omega}_{IB}(t) = \\
&\quad \underline{u}_Z\, \theta_{0_x}\, a_{SF0_y} \left\{\left[\sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right)\right] \cos\left(\Omega_y\, t - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right. \\
&\quad \left. - \frac{1}{\Omega_y}\left[\sin\left(\Omega_y\, t - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega_y\, t_0 - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right] \Omega_x \cos\left(\Omega_x\, t - \varphi_{\theta_x}\right)\right\} \qquad (10.1.2.1\text{-}6) \\
&= \underline{u}_Z\, \theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega_y}\left\{\left[\sin\left(\Omega_x\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x\, t_0 - \varphi_{\theta_x}\right)\right] \Omega_y \cos\left(\Omega_y\, t - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right. \\
&\quad \left. - \left[\sin\left(\Omega_y\, t - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega_y\, t_0 - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right] \Omega_x \cos\left(\Omega_x\, t - \varphi_{\theta_x}\right)\right\}
\end{aligned}$$

Comparing Equation (10.1.2.1-6) with (10.1.1.1-5), we see that they are identical in form; (10.1.1.1-5) can be converted to (10.1.2.1-6) by substituting $\theta_{0_y} \rightarrow a_{SF0_y} \frac{1}{\Omega_y}$ and $\varphi_{\theta_y} \rightarrow \varphi_{a_{SF_y}} + \frac{\pi}{2}$. This same substitution can then be used in (10.1.1.1-7) with (10.1.1.1-8) (both derived from (10.1.1.1-5)) to quickly determine an analytical expression for the integral of (10.1.2.1-6):

$$\int_{t_0}^{t} \frac{1}{2}\left(\underline{\mathcal{A}}(\tau) \times \underline{a}_{SF}(\tau) + \underline{\mathcal{V}}(\tau) \times \underline{\omega}_{IB}(\tau)\right) d\tau = \tag{10.1.2.1-7}$$

$$\begin{aligned}
&\underline{u}_Z \frac{1}{2}\theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega_y}\left\{\sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) - \frac{\pi}{2}\right] \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t - t_0\right)\right]\right. \\
&- \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right) - \frac{\pi}{2}\right] \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)\right]
\end{aligned}$$

(Continued)

$$
\begin{aligned}
&- \frac{\left(\Omega_y - \Omega_x\right)}{2\left(\Omega_y + \Omega_x\right)} \left(\cos\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right) - \frac{\pi}{2}\right] - \cos\left[\left(\Omega_y + \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right) - \frac{\pi}{2}\right]\right) \\
&+ \frac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)} \left(\cos\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) - \frac{\pi}{2}\right] - \cos\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) - \frac{\pi}{2}\right]\right) \Bigg\}
\end{aligned}
$$

(10.1.2.1-7)
(Continued)

or, equivalently:

$$
\int_{t_0}^{t} \frac{1}{2} \left(\underline{\mathcal{A}}(\tau) \times \underline{a}_{SF}(\tau) + \underline{\mathcal{V}}(\tau) \times \underline{\omega}_{IB}(\tau)\right) d\tau =
$$

$$
\begin{aligned}
&\underline{u}_Z \frac{1}{2} \theta_{0_x} a_{SF0_y} \frac{1}{\Omega_y} \Bigg\{ - \cos\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t - t_0\right)\right] \\
&+ \cos\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)\right] \qquad (10.1.2.1\text{-}8) \\
&- \frac{\left(\Omega_y - \Omega_x\right)}{2\left(\Omega_y + \Omega_x\right)} \left(\sin\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right)\right] - \sin\left[\left(\Omega_y + \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right)\right]\right) \\
&+ \frac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)} \left(\sin\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] - \sin\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right]\right) \Bigg\}
\end{aligned}
$$

Finally, we substitute (10.1.2.1-8), (10.1.2.1-5) and (10.1.2.1-4) in (10.1.2.1-3) to obtain for the vibration induced velocity $\underline{v}_{SF}(t)$:

$$
\begin{aligned}
&\underline{v}_{SF}(t) = \underline{u}_y a_{SF0_y} \frac{1}{\Omega_y} \left[\cos\left(\Omega_y t_0 - \varphi_{a_{SF_y}}\right) - \cos\left(\Omega_y t - \varphi_{a_{SF_y}}\right)\right] \\
&+ \underline{u}_Z \frac{1}{2} \theta_{0_x} a_{SF0_y} \frac{1}{\Omega_y} \Bigg\{ \left[\sin\left(\Omega_x t - \varphi_{\theta_x}\right) - \sin\left(\Omega_x t_0 - \varphi_{\theta_x}\right)\right] \left[\cos\left(\Omega_y t_0 - \varphi_{a_{SF_y}}\right) - \cos\left(\Omega_y t - \varphi_{a_{SF_y}}\right)\right] \\
&- \cos\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t - t_0\right)\right] \\
&+ \cos\left[\frac{1}{2}\left(\Omega_y + \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)\right] \qquad (10.1.2.1\text{-}9) \\
&- \frac{\left(\Omega_y - \Omega_x\right)}{2\left(\Omega_y + \Omega_x\right)} \left(\sin\left[\left(\Omega_y + \Omega_x\right) t - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right)\right] - \sin\left[\left(\Omega_y + \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} + \varphi_{\theta_x}\right)\right]\right) \\
&+ \frac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)} \left(\sin\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] - \sin\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right]\right) \Bigg\}
\end{aligned}
$$

Equation (10.1.2.1-9) shows that for the hypothesized Equation (10.1.2-2) and (10.1.2-3) sinusoidal X/Y angular-rate/linear-acceleration vibration profile, the velocity response is also sinusoidal at the acceleration frequency Ω_y along B Frame axis Y. Along B Frame axis Z, however, the response is a constant plus sinusoids at the angular rate and acceleration frequencies Ω_x and Ω_y (from expansion of the leading square bracketed product term), plus sinusoids at the sum and difference frequencies $\left(\Omega_y + \Omega_x\right)$, $\left(\Omega_y - \Omega_x\right)$ (as shown explicitly in (10.1.2.1-9) but also including a term from expansion of the leading square bracketed product). Hence, the velocity response has no net build-up, provided that the frequency difference $\left(\Omega_y - \Omega_x\right)$ is non-zero (i.e., $\Omega_x \neq \Omega_y$). For the case when Ω_x approaches Ω_y, the frequency difference sine functions become very low in frequency, approaching a constant, and the associated multiplication term $\dfrac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)}$ becomes large. In the limit, when Ω_x and Ω_y are equal, the product of $\dfrac{\left(\Omega_y + \Omega_x\right)}{2\left(\Omega_y - \Omega_x\right)}$ with the frequency difference sines produces a linearly increasing value for the Z component of $\underline{v}_{SF}(t)$ as the following analysis shows.

For the case when Ω_x approaches Ω_y and in the limit, equals Ω_y, Equation (10.1.2.1-9) becomes:

For $\Omega_x \;\rightarrow\; \Omega_y \;=\; \Omega$:

$$
\begin{aligned}
\underline{v}_{SF}(t) \;=\; & \underline{u}_y\, a_{SF0_y} \frac{1}{\Omega}\left[\cos\left(\Omega\, t_0 - \varphi_{a_{SF_y}}\right) - \cos\left(\Omega\, t - \varphi_{a_{SF_y}}\right)\right] \\
& + \underline{u}_z\, \frac{1}{2}\,\theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega}\left\{\left[\sin\left(\Omega\, t - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right)\right]\left[\cos\left(\Omega\, t_0 - \varphi_{a_{SF_y}}\right)\right.\right. \\
& \qquad \left. - \cos\left(\Omega\, t - \varphi_{a_{SF_y}}\right)\right] - \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \sin\left(\Omega\,(t - t_0)\right) \\
& \quad \left. + \frac{\Omega}{\left(\Omega_y - \Omega_x\right)}\left(\sin\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] - \sin\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right]\right)\right\}
\end{aligned}
\tag{10.1.2.1-10}
$$

The sine terms in Equation (10.1.2.1-10) can be reduced further as $\Omega_x \rightarrow \Omega_y = \Omega$:

$$
\begin{aligned}
& \sin\left[\left(\Omega_y - \Omega_x\right) t - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] - \sin\left[\left(\Omega_y - \Omega_x\right) t_0 - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] \\
& \quad = \; 2\cos\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t + t_0\right) - \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\right] \sin\left[\frac{1}{2}\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)\right] \\
& \quad \approx \; \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left(\Omega_y - \Omega_x\right)\left(t - t_0\right)
\end{aligned}
\tag{10.1.2.1-11}
$$

Thus, with (10.1.2.1-11) in (10.1.2.1-10), when Ω_x and Ω_y are equal, the velocity history is as follows:

$$
\begin{aligned}
\underline{v}_{SF}(t) \;=\; & \underline{u}_y \, a_{SF0_y} \frac{1}{\Omega} \left[\cos\left(\Omega \, t_0 - \varphi_{a_{SF_y}}\right) - \cos\left(\Omega \, t - \varphi_{a_{SF_y}}\right) \right] \\
+ \underline{u}_z \frac{1}{2} \theta_{0_x} \, a_{SF0_y} & \left\{ \frac{1}{\Omega} \left[\sin\left(\Omega \, t - \varphi_{\theta_x}\right) - \sin\left(\Omega \, t_0 - \varphi_{\theta_x}\right) \right] \left[\cos\left(\Omega \, t_0 - \varphi_{a_{SF_y}}\right) \right. \right. \\
& \left. \left. - \cos\left(\Omega \, t - \varphi_{a_{SF_y}}\right) \right] + \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \left[(t - t_0) - \frac{\sin\left(\Omega \, (t - t_0)\right)}{\Omega} \right] \right\}
\end{aligned}
\tag{10.1.2.1-12}
$$

Comparing Equation (10.1.2.1-12) with general Equation (10.1.2.1-9), we see that the Y axis velocity responses are sinusoidal and identical, the Z axis response has sinusoidal plus constant components, but that for $\Omega_x \rightarrow \Omega_y = \Omega$ in (10.1.2.1-12), the Z axis response contains a constant linear build-up with time $(t - t_0)$ function. The (10.1.2.1-12) Z axis linear with time build-up term has been designated as the "sculling" response which dominates the Z axis motion after one vibration cycle (i.e., $t - t_0 > 2 \, \pi \, / \, \Omega$). The term "sculling" is taken from the method used to propel a boat in the forward direction using a single oar positioned aft by the mariner and operated using an undulating motion. The undulating motion imparted to the oar generates an oscillating angular rotation of the oar shaft around the vertical while imparting thrust to the oar blade across the direction of travel when the oar shaft angle orientation is at left/right angle peaks. Due to the angle of the oar when the blade thrust is applied, a component of thrust is directed in the forward direction for both the right and left blade strokes, providing a net acceleration of the boat in the forward direction. For the vibration motion of Equations (10.1.2-1) and (10.1.2-3), the identical effect is created when the X axis angular oscillation is at the same frequency and in phase with the Y axis acceleration, creating a net acceleration along the Z axis. The linear with time coefficient in Equation (10.1.2.1-12) has been designated as the "sculling rate" and is given by:

$$
\dot{\underline{v}}_{SF_{Scul}} \;=\; \underline{u}_z \, \frac{1}{2} \, \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \tag{10.1.2.1-13}
$$

where

$\dot{\underline{v}}_{SF_{Scul}}$ = Sculling rate vector for combined B Frame X axis angular vibration and Y axis acceleration vibration at the same frequency.

Note in Equation (10.1.2.1-13), that $\dot{\underline{v}}_{SF_{Scul}}$ is proportional to the cosine of the phase angle difference between the Equations (10.1.2-1) and (10.1.2-3) X axis angular vibration and Y axis acceleration vibration components, with worst case sculling rate occurring when the angular and linear acceleration oscillations are in phase. Note also that sculling rate increases as the product of the angular and acceleration vibration amplitude products. Finally, we see from

(10.1.2.1-13) that for a given vibration angle and acceleration amplitude, the sculling rate is independent of the angular/linear vibration frequency Ω (sometimes denoted as the "sculling frequency"), so long as the angular/linear vibration frequencies are equal.

10.1.2.2 VELOCITY ALGORITHM RESPONSE

The software in a strapdown INS processes a digital integration algorithm to calculate velocity. For consistency with Section 10.1.2.1, we will assume that the velocity integration algorithm is a digital version of the continuous form Equations (10.1.2.1-2) and (10.1.2.1-3) based on attitude in the form of a rotation vector. Thus, we assume the velocity algorithm is derived from the following equivalent version of (10.1.2.1-2) and (10.1.2.1-3):

$$
\begin{aligned}
&\underline{\alpha}(t) \equiv \int_{t_{m-1}}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \underline{\upsilon}(t) \equiv \int_{t_{m-1}}^{t} \underline{a}_{SF}(\tau)\, d\tau \\
&\underline{\alpha}_m = \underline{\alpha}(t_m) \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m) \\
&\underline{\mathcal{A}}_m = \sum_{1}^{m} \underline{\alpha}_i \qquad \underline{\mathcal{V}}_m = \sum_{1}^{m} \underline{\upsilon}_i \qquad\qquad (10.1.2.2\text{-}1) \\
&\underline{\mathcal{A}}(t) = \underline{\mathcal{A}}_{m-1} + \underline{\alpha}(t) \qquad \underline{\mathcal{V}}(t) = \underline{\mathcal{V}}_{m-1} + \underline{\upsilon}(t) \\
&\underline{v}_{SF_m} = \underline{\mathcal{V}}_m + \frac{1}{2}\underline{\mathcal{A}}_m \times \underline{\mathcal{V}}_m + \sum_{i=1}^{m} \int_{t_{i-1}}^{t_i} \frac{1}{2}\left(\underline{\mathcal{A}}(t) \times \underline{a}_{SF}(t) + \underline{\mathcal{V}}(t) \times \underline{\omega}_{IB}(t)\right) dt
\end{aligned}
$$

where

m = Strapdown software attitude/velocity update algorithm cycle index. The m zero cycle corresponds to time t_0 in Equations (10.1.2.1-2) and (10.1.2.1-3).

i = Particular m cycle over the m cycle history.

$\underline{\alpha}(t)$ = Integrated B Frame angular rate from the last m cycle time t_{m-1} to time t.

$\underline{\upsilon}(t)$ = Integrated B Frame specific force from the last m cycle time t_{m-1} to time t.

$\underline{\alpha}_m$, $\underline{\upsilon}_m$ = Integrated B Frame angular rate and specific force from t_{m-1} to time t_m.

$\underline{v}_{SF_m}$ = Velocity due to angular/linear vibration in the t_0 B Frame at computer cycle m.

The integral term in the $\underline{v}_{SF_m}$ expression can be expanded if we substitute for $\underline{\mathcal{A}}(t)$, $\underline{\mathcal{V}}(t)$ and apply the $\underline{\alpha}_m$, $\underline{\upsilon}_m$ definition from (10.1.2.2-1):

$$\int_{t_{i-1}}^{t_i} \frac{1}{2}\left(\underline{\mathcal{A}}(t) \times \underline{a}_{SF}(t) + \underline{\mathcal{V}}(t) \times \underline{\omega}_{IB}(t)\right) dt$$

$$= \int_{t_{i-1}}^{t_i} \frac{1}{2}\left[\left(\underline{\mathcal{A}}_{i-1} + \underline{\alpha}(t)\right) \times \underline{a}_{SF}(t) + \left(\underline{\mathcal{V}}_{i-1} + \underline{\upsilon}(t)\right) \times \underline{\omega}_{IB}(t)\right] dt$$

$$= \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right) + \int_{t_{i-1}}^{t_i} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{a}_{SF}(t) + \underline{\upsilon}(t) \times \underline{\omega}_{IB}(t)\right) dt \qquad (10.1.2.2\text{-}2)$$

$$= \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right) + \Delta\underline{v}_{Scul_i}$$

with

$$\Delta\underline{v}_{Scul_i} \equiv \int_{t_{i-1}}^{t_i} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{a}_{SF}(t) + \underline{\upsilon}(t) \times \underline{\omega}_{IB}(t)\right) dt \qquad (10.1.2.2\text{-}3)$$

where

$\Delta\underline{v}_{Scul_i}$ = Sculling contribution to integral in (10.1.2.2-1).

With (10.1.2.2-2) and (10.1.2.2-3), Equations (10.1.2.2-1) become:

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \underline{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{a}_{SF}(\tau)\, d\tau$$

$$\Delta\underline{v}_{Scul_i} \equiv \int_{t_{i-1}}^{t_i} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{a}_{SF}(t) + \underline{\upsilon}(t) \times \underline{\omega}_{IB}(t)\right) dt$$

$$\underline{\alpha}_m = \underline{\alpha}(t_m) \qquad \underline{\upsilon}_m = \underline{\upsilon}(t_m) \qquad (10.1.2.2\text{-}4)$$

$$\underline{\mathcal{A}}_m = \sum_1^m \underline{\alpha}_i \qquad \underline{\mathcal{V}}_m = \sum_1^m \underline{\upsilon}_i$$

$$\underline{v}_{SF_m} = \underline{\mathcal{V}}_m + \frac{1}{2}\underline{\mathcal{A}}_m \times \underline{\mathcal{V}}_m + \sum_1^m \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right) + \sum_1^m \Delta\underline{v}_{Scul_i}$$

Equations (10.1.2.2-4) represent the equivalent digital forms of Equations (10.1.2.1-2) and (10.1.2.1-3), generating the identical solution at the m cycle times. The $\underline{\alpha}$, $\underline{\upsilon}$, $\Delta\underline{v}_{Scul}$ terms in (10.1.2.2-4) should be recognized as the identical terms in Section 7.2.2.2 for strapdown velocity algorithm development. The following subsections analyze the response of Equations

(10.1.2.2-4) under the hypothesized Section 10.1.2 sinusoidal angular/linear vibrations for the exact implementation of (10.1.2.2-4) and for the approximate forms typified in the INS software.

10.1.2.2.1 Exact Velocity Algorithm Response

The (10.1.2.2-4) algorithm is exact, hence, its overall response at the m cycle times to general X, Y sinusoidal angular/linear vibration is identical to (10.1.2.1-9) which was derived from (10.1.2.1-2) - (10.1.2.1-3). Similarly, the overall response for equal X, Y angular/linear vibration frequencies is identical to the (10.1.2.1-12) - (10.1.2.1-13) results. It is instructive to analyze the individual contributions in (10.1.2.2-4) to the overall result. For this discussion, we will only address the case when the X, Y angular vibration frequencies are equal (i.e., $\Omega_x = \Omega_y = \Omega$).

From (10.1.2.1-4) (the exact equivalent to the Equations (10.1.2.2-4) $\underline{\mathcal{A}}_m$, $\underline{\mathcal{V}}_m$ expressions at t_m), we can write for $\underline{\mathcal{A}}_m$, $\underline{\mathcal{V}}_m$ under the (10.1.2-2) and (10.1.2-4) vibration exposure with $\Omega_x = \Omega_y = \Omega$:

$$\begin{aligned}\underline{\mathcal{A}}_m &= \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_m - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right)\right] \\ \underline{\mathcal{V}}_m &= \underline{u}_y\, a_{SF0_y} \frac{1}{\Omega}\left[\sin\left(\Omega\, t_m - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right]\end{aligned} \qquad (10.1.2.2.1\text{-}1)$$

Using (10.1.2.2.1-1) we find:

$$\begin{aligned}\frac{1}{2}\underline{\mathcal{A}}_m \times \underline{\mathcal{V}}_m = \underline{u}_z \frac{1}{2}\theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega}\Big[\sin\left(\Omega\, t_m - \varphi_{\theta_x}\right) \\ - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right)\Big]\left[\cos\left(\Omega\, t_0 - \varphi_{a_{SF_y}}\right) - \cos\left(\Omega\, t_m - \varphi_{a_{SF_y}}\right)\right]\end{aligned} \qquad (10.1.2.2.1\text{-}2)$$

From the definition for $\underline{\alpha}_m$, $\underline{\upsilon}_m$ in (10.1.2.2-4), we can also write for the (10.1.2-2) and (10.1.2-4) vibration at $\Omega_x = \Omega_y = \Omega$:

$$\underline{\alpha}_m = \int_{t_{m-1}}^{t_m} \underline{\omega}_{IB}(t)\, dt = \underline{u}_x\, \theta_{0_x}\left[\sin\left(\Omega\, t_m - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{m-1} - \varphi_{\theta_x}\right)\right] \qquad (10.1.2.2.1\text{-}3)$$

$$\underline{\upsilon}_m = \int_{t_{m-1}}^{t_m} \underline{a}_{SF}(t)\, dt = \underline{u}_y\, a_{SF0_y}\frac{1}{\Omega}\left[\sin\left(\Omega\, t_m - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_{m-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right]$$

The combination of (10.1.2.2.1-1) and (10.1.2.2.1-3) for m = i then yields an expression for the cross-product term under the summation in the (10.1.2.2-4) $\underline{v}_{SF_m}$ equation:

$$\begin{aligned}\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i \; = \; & \underline{u}_Z \, \theta_{0_x} \, a_{SF0_y} \frac{1}{\Omega} \Bigg\{ \Big[\sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right) \\ & - \sin\left(\Omega\, t_0 - \varphi_{\theta_x}\right)\Big] \left[\sin\left(\Omega\, t_i - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_{i-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right] \\ - & \left[\sin\left(\Omega\, t_{i-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right] \Big[\sin\left(\Omega\, t_i - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{i-1} - \varphi_{\theta_x}\right)\Big] \Bigg\}\end{aligned} \qquad (10.1.2.2.1\text{-}4)$$

If we compare Equation (10.1.2.2.1-4) with (10.1.1.2.1-3) we see that they are identical in form; (10.1.1.2.1-3) can be converted to (10.1.2.2.1-4) by substituting $\theta_{0_y} \rightarrow a_{SF0_y} \frac{1}{\Omega}$ and $\varphi_{\theta_y} \rightarrow \varphi_{a_{SF_y}} + \frac{\pi}{2}$. This same substitution can then be used in (10.1.1.2.1-8) (which was derived from (10.1.1.2.1-3)) to quickly determine an analytical expression for the $\sum_1^m \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right)$ term in (10.1.2.2-4):

$$\begin{aligned}&\sum_1^m \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right) \; = \\ &\quad \underline{u}_Z \frac{1}{2} \theta_{0_x} a_{SF0_y} \sin\left(\varphi_{a_{SF_y}} + \frac{\pi}{2} - \varphi_{\theta_x}\right)\left(\frac{\sin\, \Omega\, T_m}{\Omega\, T_m}\left(t_m - t_0\right) - \frac{\sin\, \Omega\left(t_m - t_0\right)}{\Omega}\right) \\ &\quad = \; \underline{u}_Z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left(\frac{\sin\, \Omega\, T_m}{\Omega\, T_m}\left(t_m - t_0\right) - \frac{\sin\, \Omega\left(t_m - t_0\right)}{\Omega}\right)\end{aligned} \qquad (10.1.2.2.1\text{-}5)$$

An analytical expression for the sculling term in $\underline{v}_{SF_m}$ Equation (10.1.2.2-4) can be easily obtained by direct extension of Equation (10.1.2.1-12), the solution to (10.1.2.1-3) under (10.1.2-2) and (10.1.2-4) vibration with $\Omega_x = \Omega_y = \Omega$. On review of the (10.1.2.1-12) derivation, it should be clear that the second part of the (10.1.2.1-12) Z component is the integral term in (10.1.2.1-3), or with (10.1.2.1-2):

$$\underline{\mathcal{A}}(t) = \int_{t_0}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \underline{\mathcal{V}}(t) = \int_{t_0}^{t} \underline{a}_{SF}(\tau)\, d\tau$$

$$\int_{t_0}^{t_m} \frac{1}{2}\left(\underline{\mathcal{A}}(t) \times \underline{a}_{SF}(t) + \underline{\mathcal{V}}(t) \times \underline{\omega}_{IB}(t)\right) dt = \tag{10.1.2.2.1-6}$$

$$\underline{u}_Z \frac{1}{2}\, \theta_{0_x}\, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left[\left(t_m - t_0\right) - \frac{\sin\left(\Omega\,(t_m - t_0)\right)}{\Omega}\right]$$

For comparison, the calculations for $\Delta\underline{v}_{Scul_m}$ in Equations (10.1.2.2-4) are repeated below:

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}(\tau)\, d\tau \qquad \underline{\upsilon}(t) = \int_{t_{m-1}}^{t} \underline{a}_{SF}(\tau)\, d\tau$$

$$\Delta\underline{v}_{Scul_m} = \int_{t_{m-1}}^{t_m} \frac{1}{2}\left(\underline{\alpha}(t) \times \underline{a}_{SF}(t) + \underline{\upsilon}(t) \times \underline{\omega}_{IB}(t)\right) dt \tag{10.1.2.2.1-7}$$

Comparing Equations (10.1.2.2.1-6) and (10.1.2.2.1-7) we see that they are analytically equivalent; (10.1.2.2.1-6) can be converted to (10.1.2.2.1-7) through the following substitutions:

$$\underline{\mathcal{A}}(t) \rightarrow \underline{\alpha}(t) \qquad \underline{\mathcal{V}}(t) \rightarrow \underline{\upsilon}(t) \qquad t_0 \rightarrow t_{m-1} \qquad t_m - t_{m-1} = T_m \tag{10.1.2.2.1-8}$$

for which the $\Delta\underline{v}_{Scul_m}$ analytical solution becomes:

$$\Delta\underline{v}_{Scul_m} = \underline{u}_Z \frac{1}{2}\, \theta_{0_x}\, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) T_m \tag{10.1.2.2.1-9}$$

Thus, $\Delta\underline{v}_{Scul_m}$ is constant and the $\Delta\underline{v}_{Scul_i}$ summation term in the (10.1.2.2-4) $\underline{v}_{SF_m}$ expression is given by:

$$\begin{aligned}\sum_{1}^{m} \Delta\underline{v}_{Scul_i} &= m\, \Delta\underline{v}_{Scul_m} = \frac{\left(t_m - t_0\right)}{T_m}\, \Delta\underline{v}_{Scul_m} \\ &= \underline{u}_Z \frac{1}{2}\, \theta_{0_x}\, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)\left(t_m - t_0\right)\end{aligned} \tag{10.1.2.2.1-10}$$

Because $\Delta \underline{v}_{Scul_m}$ is constant, we can also define a $\underline{v}_{SF_m}$ velocity build-up rate contribution associated with the $\Delta \underline{v}_{Scul_i}$ sculling summation term in (10.1.2.2-4) as the $\left(t_m - t_0\right)$ coefficient in (10.1.2.2.1-10):

$$\Delta \dot{\underline{v}}_{Scul_m} = \underline{u}_Z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) \tag{10.1.2.2.1-11}$$

where

$\Delta \dot{\underline{v}}_{Scul_m}$ = Constant rate of change in $\underline{v}_{SF_m}$ generated by the summing of $\Delta \underline{v}_{Scul_i}$'s in Equation (10.1.2.2-4).

As an exercise, we form $\underline{v}_{SF_m}$ in (10.1.2.2-4) by combining (10.1.2.2.1-10), (10.1.2.2.1-5), (10.1.2.2.1-2) and the $\underline{\mathcal{V}}_m$ expression in (10.1.2.2.1-1):

$$\begin{aligned}
\underline{v}_{SF_m} = \; & \underline{u}_y \, a_{SF0_y} \frac{1}{\Omega}\left[\cos\left(\Omega \, t_0 - \varphi_{a_{SF_y}}\right) - \cos\left(\Omega \, t_m - \varphi_{a_{SF_y}}\right)\right] \\
+ \; & \underline{u}_Z \frac{1}{2} \theta_{0_x} a_{SF0_y} \left\{\frac{1}{\Omega}\left[\sin\left(\Omega \, t_m - \varphi_{\theta_x}\right) - \sin\left(\Omega \, t_0 - \varphi_{\theta_x}\right)\right]\left[\cos\left(\Omega \, t_0 - \varphi_{a_{SF_y}}\right)\right.\right. \\
& \left.\left. - \cos\left(\Omega \, t_m - \varphi_{a_{SF_y}}\right)\right] + \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left[\left(t_m - t_0\right) - \frac{\sin\left(\Omega\left(t_m - t_0\right)\right)}{\Omega}\right]\right\}
\end{aligned} \tag{10.1.2.2.1-12}$$

The result is identical to (10.1.2.1-12) at the m cycle times (as it should since (10.1.2.2-4) is an exact algorithm).

Let us now review the results of our findings. In Section 10.1.2.1 we noted that for time greater than one vibration cycle, the $\left(t_m - t_0\right)$ term in (10.1.2.2.1-12) dominates the $\underline{v}_{SF_m}$ Z axis response, having a linear with time "sculling rate" build-up with slope:

$$\dot{\underline{v}}_{SF_{Scul}} = \underline{u}_Z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \tag{10.1.2.2.1-13}$$

Comparing $\Delta \dot{\underline{v}}_{Scul_m}$ in Equation (10.1.2.2.1-11) and $\dot{\underline{v}}_{SF_{Scul}}$ in (10.1.2.2.1-13), we see that:

$$\Delta \dot{\underline{v}}_{Scul_m} = \left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) \dot{\underline{v}}_{SF_{Scul}} \tag{10.1.2.2.1-14}$$

Thus, $\Delta\dot{\underline{v}}_{Scul_m}$ measures the $\left(1 - \dfrac{\sin \Omega T_m}{\Omega T_m}\right)$ portion of $\dot{\underline{v}}_{SF_{Scul}}$. For large ΩT_m (i.e., large sculling rate compared to the attitude/velocity update frequency $1 / T_m$), the $\dfrac{\sin \Omega T_m}{\Omega T_m}$ term in (10.1.2.2.1-14) goes to zero and $\Delta\dot{\underline{v}}_{Scul_m}$ becomes equal to the full sculling rate $\dot{\underline{v}}_{SF_{Scul}}$. For small ΩT_m, the $\dfrac{\sin \Omega T_m}{\Omega T_m}$ term goes to one, $\left(1 - \dfrac{\sin \Omega T_m}{\Omega T_m}\right)$ goes to zero, hence, $\Delta\dot{\underline{v}}_{Scul_m}$ goes to zero. Therefore, $\Delta\dot{\underline{v}}_{Scul_m}$ measures the high frequency portion of the total sculling rate (i.e., high frequency compared to the attitude/velocity update frequency).

The total velocity is formed in the (10.1.2.2-4) $\underline{v}_{SF_m}$ equation by the summing of $\underline{\mathcal{V}}_m$ from (10.1.2.2.1-1), $\sum_1^m \Delta\underline{v}_{Scul_i}$ from (10.1.2.2.1-10), $\frac{1}{2}\underline{\mathcal{A}}_m \times \underline{\mathcal{V}}_m$ from (10.1.2.2.1-2) and $\sum_1^m \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right)$ from (10.1.2.2.1-5). The sculling contribution to $\underline{v}_{SF_m}$ is manifested in the $\dot{\underline{v}}_{SF_{Scul}}$ build-up rate of Equation (10.1.2.2.1-13). From (10.1.2.2.1-1) we see that $\underline{\mathcal{V}}_m$ has no contribution to the sculling rate which is about $\underline{u}_Z$. The contribution of $\sum_1^m \Delta\underline{v}_{Scul_i}$ to the $\underline{v}_{SF_m}$ sculling rate is $\Delta\dot{\underline{v}}_{Scul_m}$ in (10.1.2.2.1-14) which, as discussed previously, measures the $\left(1 - \dfrac{\sin \Omega T_m}{\Omega T_m}\right)$ high frequency portion of $\dot{\underline{v}}_{SF_{Scul}}$. From (10.1.2.2.1-5) we see that the $\dfrac{\sin \Omega T_m}{\Omega T_m}$ portion of $\dot{\underline{v}}_{SF_{Scul}}$ (i.e., the portion not measured by $\Delta\dot{\underline{v}}_{Scul_m}$) is contained in the $\sum_1^m \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right)$ term. Thus $\sum_1^m \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_i + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_i\right)$ in (10.1.2.2-4) measures the low frequency portion of $\dot{\underline{v}}_{SF_{Scul}}$.

A review of Section 7.2.2.2 reveals that $\Delta\underline{v}_{Scul_m}$ in (10.1.2.2-4) is the identical sculling term measured by the high speed part of the two-speed velocity update algorithm (see Equations (7.2.2.2-23) - (7.2.2.2-25)). The low speed (m cycle) part of the velocity update algorithm measures the remaining $\underline{\mathcal{V}}_{m} + \frac{1}{2}\,\underline{\mathcal{A}}_{m} \times \underline{\mathcal{V}}_{m} + \sum_{1}^{m} \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_{i} + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_{i}\right)$ term in (10.1.2.2-4). Thus, from (10.1.2.2.1-14), $\Delta\dot{\underline{v}}_{Scul_m}$ for the high speed algorithm measures $\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)$ of the total sculling rate, while the remaining $\frac{\sin \Omega T_m}{\Omega T_m}$ sculling rate portion is measured by the low speed part of the two-speed algorithm.

Equation (10.1.2.2.1-14) is based on an analytical integration for $\Delta\underline{v}_{Scul_m}$ in (10.1.2.2-4). In this sense, (10.1.2.2.1-14) can be considered as the solution for the "exact" sculling algorithm for which exact denotes an infinitely fast computer capable of executing the continuous analytical integration operation. The low speed portion of the two-speed velocity updating algorithm was derived in Section 7.2.2.2 as an exact solution (assuming an exact input from the high speed portion). Thus, $\underline{\mathcal{V}}_{m} + \frac{1}{2}\,\underline{\mathcal{A}}_{m} \times \underline{\mathcal{V}}_{m} + \sum_{1}^{m} \frac{1}{2}\left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_{i} + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_{i}\right)$ in (10.1.2.2-4) represents the low speed portion of both the exact version, and the Section 7.2.2.2 INS computer velocity updating algorithm version.

10.1.2.2.2 <u>INS Velocity Algorithm Response And Error</u>

In the last section we analyzed the response of an exact velocity computation algorithm to hypothesized sinusoidal vibrations to discriminate between the total velocity solution and the portion of the velocity solution contributed by the exact $\Delta\underline{v}_{Scul_m}$ sculling algorithm. We showed that under combined angular/linear vibrations of identical frequency around the B Frame X axis and along the B Frame Y axis, the overall velocity developed an unbounded linear increase with time along the B Frame Z axis at a constant total "sculling rate" $\dot{\underline{v}}_{SF_{Scul}}$ given by Equation (10.1.2.2.1-13). In this section we will analyze the response of the velocity algorithm implemented in the strapdown INS which we model after the Equation (10.1.2.2-4) general form:

$$\underline{\mathcal{A}}_{Algo_m} = \sum_{1}^{m} \underline{\alpha}_{Algo_i} \qquad \underline{\mathcal{V}}_{Algo_m} = \sum_{1}^{m} \underline{\upsilon}_{Algo_i}$$

$$\underline{v}_{SF/Algo_m} = \underline{\mathcal{V}}_{Algo_m} + \frac{1}{2}\,\underline{\mathcal{A}}_{Algo_m} \times \underline{\mathcal{V}}_{Algo_m} \qquad (10.1.2.2.2\text{-}1)$$
$$+ \sum_{1}^{m} \frac{1}{2}\left(\underline{\mathcal{A}}_{Algo\,i-1} \times \underline{\upsilon}_{Algo\,i} + \underline{\mathcal{V}}_{Algo\,i-1} \times \underline{\alpha}_{Algo\,i}\right) + \sum_{1}^{m} \Delta\underline{v}_{Scul/Algo_i}$$

where

$(\,)_{\text{Algo}}$ = Version of ( ) in Equations (10.1.2.2-4) implemented in the strapdown INS software.

Let us further assume that the (10.1.2.2.2-1) INS software attitude algorithm is of the Section 7.2.2.2 two-speed type, with $\underline{\mathcal{V}}_{\text{Algo}_\text{m}} + \frac{1}{2}\,\underline{\mathcal{A}}_{\text{Algo}_\text{m}} \times \underline{\mathcal{V}}_{\text{Algo}_\text{m}} + \sum_{1}^{m} \frac{1}{2}\left(\underline{\mathcal{A}}_{\text{Algo}_{\text{i-1}}} \times \underline{\upsilon}_{\text{Algo}_\text{i}} + \underline{\mathcal{V}}_{\text{Algo}_{\text{i-1}}} \times \underline{\alpha}_{\text{Algo}_\text{i}}\right)$ implemented as an exact low-speed algorithm (as in Section 7.2.2.2), and the high speed portion implemented as in Equations (7.2.2.2.2-13) - (7.2.2.2.2-15):

$$\Delta\underline{\alpha}_{\text{Algo}_l} = \int_{t_{l-1}}^{t_l} d\underline{\alpha} \qquad \begin{array}{c}\text{Summation Of Integrated Angular Rate Output}\\ \text{Increments From Angular Rate Sensors}\end{array}$$

$$\underline{\alpha}_{\text{Algo}_l} = \underline{\alpha}_{\text{Algo}_{l-1}} + \Delta\underline{\alpha}_{\text{Algo}_l} \tag{10.1.2.2.2-2}$$

$$\underline{\alpha}_{\text{Algo}_\text{m}} = \underline{\alpha}_{\text{Algo}_l}(t_l = t_m) \qquad \underline{\alpha}_{\text{Algo}_l} = 0 \quad \text{At } t = t_{m-1}.$$

$$\Delta\underline{\upsilon}_{\text{Algo}_l} = \int_{t_{l-1}}^{t_l} d\underline{\upsilon} \qquad \begin{array}{c}\text{Summation Of Integrated Specific Force}\\ \text{Output Increments From Accelerometers}\end{array}$$

$$\underline{\upsilon}_{\text{Algo}_l} = \underline{\upsilon}_{\text{Algo}_{l-1}} + \Delta\underline{\upsilon}_{\text{Algo}_l} \tag{10.1.2.2.2-3}$$

$$\underline{\upsilon}_{\text{Algo}_\text{m}} = \underline{\upsilon}_{\text{Algo}_l}(t_l = t_m) \qquad \underline{\upsilon}_{\text{Algo}_l} = 0 \quad \text{At } t = t_{m-1}.$$

$$\begin{aligned}\delta\underline{v}_{\text{Scul/Algo}_l} = \frac{1}{2}\Bigg[&\left(\underline{\alpha}_{\text{Algo}_{l-1}} + \frac{1}{6}\,\Delta\underline{\alpha}_{\text{Algo}_{l-1}}\right) \times \Delta\underline{\upsilon}_{\text{Algo}_l}\\ &+ \left(\underline{\upsilon}_{\text{Algo}_{l-1}} + \frac{1}{6}\,\Delta\underline{\upsilon}_{\text{Algo}_{l-1}}\right) \times \Delta\underline{\alpha}_{\text{Algo}_l}\Bigg]\end{aligned} \tag{10.1.2.2.2-4}$$

$$\Delta\underline{v}_{\text{Scul/Algo}_l} = \Delta\underline{v}_{\text{Scul/Algo}_{l-1}} + \delta\underline{v}_{\text{Scul/Algo}_l}$$

$$\Delta\underline{v}_{\text{Scul/Algo}_\text{m}} = \Delta\underline{v}_{\text{Scul/Algo}_l}(t_l = t_m) \qquad \Delta\underline{v}_{\text{Scul/Algo}_l} = 0 \quad \text{At } t = t_{m-1}.$$

If we compare Equations (10.1.2.2.2-1) - (10.1.2.2.2-4) with the Equations (10.1.2.2-4) "exact" algorithm equivalent, it should be apparent that the algorithms are identical except for the $\Delta\underline{v}_{\text{Scul}}$ terms; i.e.:

$$\begin{array}{lll} \Delta\underline{\alpha}_{Algo} = \Delta\underline{\alpha} & \underline{\alpha}_{Algo} = \underline{\alpha} & \underline{\mathcal{A}}_{Algo} = \underline{\mathcal{A}} \\ \Delta\underline{\upsilon}_{Algo} = \Delta\underline{\upsilon} & \underline{\upsilon}_{Algo} = \underline{\upsilon} & \underline{\mathcal{V}}_{Algo} = \underline{\mathcal{V}} \end{array} \qquad (10.1.2.2.2\text{-}5)$$

Using (10.1.2.2.2-5), we subtract the (10.1.2.2.2-1) and (10.1.2.2-4) $\underline{v}_{SF}$ total velocity expressions to obtain an equation for the INS velocity algorithm error:

$$\delta\underline{v}_{SF/Algo_m} \equiv \underline{v}_{SF/Algo_m} - \underline{v}_{SF_m} = \sum_1^m \Delta\underline{v}_{Scul/Algo_i} - \sum_1^m \Delta\underline{v}_{Scul_i} \qquad (10.1.2.2.2\text{-}6)$$

where

$\delta\underline{v}_{SF/Algo_m}$ = Total INS velocity algorithm error at completion of velocity computation cycle m.

We also define:

$$\delta\Delta\underline{v}_{Scul/Algo_m} \equiv \Delta\underline{v}_{Scul/Algo_m} - \Delta\underline{v}_{Scul_m} \qquad (10.1.2.2.2\text{-}7)$$

where

$\delta\Delta\underline{v}_{Scul/Algo_m}$ = INS sculling algorithm error for velocity computation cycle m.

Let us evaluate the individual terms in (10.1.2.2.2-6) - (10.1.2.2.2-7) under our hypothesized (10.1.2-2) and (10.1.2-4) angular-rate/linear-acceleration vibration exposure with $\Omega_x = \Omega_y = \Omega$. First, we find for the true integrated rate and acceleration terms:

$$\begin{aligned}
\underline{\alpha}_l &\equiv \int_{t_{l=(m-1)s}}^{t_l} \underline{\omega}_{IB}(t)\,dt = \underline{u}_x\,\theta_{0_x}\left[\sin\left(\Omega\,t_l - \varphi_{\theta_x}\right) - \sin\left(\Omega\,t_{l=(m-1)s} - \varphi_{\theta_x}\right)\right] \\
\underline{\upsilon}_l &\equiv \int_{t_{l=(m-1)s}}^{t_l} \underline{a}_{SF}(t)\,dt = \underline{u}_y\,a_{SF0_y}\,\frac{1}{\Omega}\left[\sin\left(\Omega\,t_l - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\,t_{l=(m-1)s} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right] \\
\Delta\underline{\alpha}_l &= \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}(t)\,dt = \underline{u}_x\,\theta_{0_x}\left[\sin\left(\Omega\,t_l - \varphi_{\theta_x}\right) - \sin\left(\Omega\,t_{l-1} - \varphi_{\theta_x}\right)\right] \qquad (10.1.2.2.2\text{-}8) \\
\Delta\underline{\upsilon}_l &\equiv \int_{t_{l-1}}^{t_l} \underline{a}_{SF}(t)\,dt = \underline{u}_y\,a_{SF0_y}\,\frac{1}{\Omega}\left[\sin\left(\Omega\,t_l - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\,t_{l-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right)\right]
\end{aligned}$$

where

s = Number of l cycles in an m cycle.

We then find an analytical expression for $\Delta \underline{v}_{Scul/Algo_m}$ in (10.1.2.2.2-4) which we first rewrite using (10.1.2.2.2-5):

$$\begin{aligned} \Delta \underline{v}_{Scul/Algo_m} = & \sum_{(m-1)s+1}^{ms} \frac{1}{2} \left(\underline{\alpha}_{l-1} \times \Delta \underline{\upsilon}_{l} + \underline{\upsilon}_{l-1} \times \Delta \underline{\alpha}_{l} \right) \\ & + \sum_{(m-1)s+1}^{ms} \frac{1}{12} \left(\Delta \underline{\alpha}_{l-1} \times \Delta \underline{\upsilon}_{l} + \Delta \underline{\upsilon}_{l-1} \times \Delta \underline{\alpha}_{l} \right) \end{aligned} \qquad (10.1.2.2.2\text{-}9)$$

Following the procedure that led to $\sum_{1}^{m} \frac{1}{2} \left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_{i} + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_{i} \right)$ in Equation (10.1.2.2.1-5) from (10.1.2.2.1-1) and (10.1.2.2.1-3), Equations (10.1.2.2.2-8) can be combined to derive an analytical expression for the $\sum_{(m-1)s+1}^{ms} \frac{1}{2} \left(\underline{\alpha}_{l-1} \times \Delta \underline{\upsilon}_{l} + \underline{\upsilon}_{l-1} \times \Delta \underline{\alpha}_{l} \right)$ term in (10.1.2.2.2-9). Alternatively, $\sum_{(m-1)s+1}^{ms} \frac{1}{2} \left(\underline{\alpha}_{l-1} \times \Delta \underline{\upsilon}_{l} + \underline{\upsilon}_{l-1} \times \Delta \underline{\alpha}_{l} \right)$ can be determined by inspection if we compare (10.1.2.2.2-8) with Equations (10.1.2.2.1-1) and (10.1.2.2.1-3), and noting that they are of identical form. This observation allows us to quickly write $\sum_{(m-1)s+1}^{ms} \frac{1}{2} \left(\underline{\alpha}_{l-1} \times \Delta \underline{\upsilon}_{l} + \underline{\upsilon}_{l-1} \times \Delta \underline{\alpha}_{l} \right)$ from the $\sum_{1}^{m} \frac{1}{2} \left(\underline{\mathcal{A}}_{i-1} \times \underline{\upsilon}_{i} + \underline{\mathcal{V}}_{i-1} \times \underline{\alpha}_{i} \right)$ result if we translate the (10.1.2.2.1-5) terms as:

$$\underline{\mathcal{A}}_{i-1} \rightarrow \underline{\alpha}_{l-1} \qquad \underline{\alpha}_{i} \rightarrow \Delta \underline{\alpha}_{l} \qquad \underline{\mathcal{V}}_{i-1} \rightarrow \underline{\upsilon}_{l-1} \qquad \underline{\upsilon}_{i} \rightarrow \Delta \underline{\upsilon}_{l} \qquad T_m \rightarrow T_l \qquad t_m \rightarrow t_{l=ms}$$

$$t_0 \rightarrow t_{l=(m-1)s} \qquad \left(t_m - t_0 \right) \rightarrow \left(t_{l=ms} - t_{l=(m-1)s} \right) = s\, T_l = T_m$$

$$(i = 1 \text{ to } m) \rightarrow \left[l = (m-1)s+1 \text{ to } ms \right]$$

Then we quickly find:

$$\begin{aligned} & \sum_{(m-1)s+1}^{ms} \frac{1}{2} \left(\underline{\alpha}_{l-1} \times \Delta \underline{\upsilon}_{l} + \underline{\upsilon}_{l-1} \times \Delta \underline{\alpha}_{l} \right) = \\ & \qquad \underline{u}_z \, \frac{1}{2} \, \theta_{0_x} \, a_{SF0_y} \cos \left(\varphi_{a_{SF_y}} - \varphi_{\theta_x} \right) \left(\frac{\sin \Omega T_l}{\Omega T_l} - \frac{\sin \Omega T_m}{\Omega T_m} \right) T_m \end{aligned} \qquad (10.1.2.2.2\text{-}10)$$

The $\sum_{(m-1)s+1}^{ms} \frac{1}{12}\left(\Delta\underline{\alpha}_{\,l-1} \times \Delta\underline{\upsilon}_{\,l} + \Delta\underline{\upsilon}_{\,l-1} \times \Delta\underline{\alpha}_{\,l}\right)$ term in (10.1.2.2.2-9) is unique to the particular Section 7.2.2.2.2 second order sculling algorithm. A more general higher order form of (10.1.2.2.2-9) for sculling algorithms based on *l* cycle integrated angular-rate/linear-acceleration measurements is:

$$
\begin{aligned}
&\Delta\underline{\alpha}_j \equiv \int_{t_{j-1}}^{t_j} d\underline{\alpha} \qquad \Delta\underline{\alpha}_k \equiv \int_{t_{k-1}}^{t_k} d\underline{\alpha} \\
&\Delta\underline{\upsilon}_j \equiv \int_{t_{j-1}}^{t_j} d\underline{\upsilon} \qquad \Delta\underline{\upsilon}_k \equiv \int_{t_{k-1}}^{t_k} d\underline{\upsilon} \\
&\Delta\underline{v}_{Scul/Algo_m} = \sum_{(m-1)s+1}^{ms} \frac{1}{2}\left(\underline{\alpha}_{\,l-1} \times \Delta\underline{\upsilon}_{\,l} + \underline{\upsilon}_{\,l-1} \times \Delta\underline{\alpha}_{\,l}\right) \\
&\qquad + \sum_{l=(m-1)s+1}^{ms} \sum_{j,\,k} C_{j,k}\left(\Delta\underline{\alpha}_j \times \Delta\underline{\upsilon}_k + \Delta\underline{\upsilon}_j \times \Delta\underline{\alpha}_k\right)
\end{aligned}
\tag{10.1.2.2.2-11}
$$

where

j, k = High speed computer cycle indices at the *l* cycle rate that are displaced from the *l* cycle by integers.

$C_{j,k}$ = Coefficient for the j, k product.

For the Section 7.2.2.2.2 second order sculling algorithm represented by (10.1.2.2.2-9), the j, k, and $C_{j,k}$ values are $j = l\text{-}1$, $k = l$ and $C_{j,k} = 1/12$. Variations on (10.1.2.2.2-11) are also possible (e.g., Reference 13) in which the j, k computer cycle rate is faster than the *l* cycle rate by an integer multiple with the j, k summation performed once each *l* cycle. This latter approach requires the angular rate sensor and accelerometer $\Delta\underline{\alpha}$, $\Delta\underline{\upsilon}$ summers to be sampled at the faster j, k cycle rate. Returning to our original j, k definition (i.e., at the *l* cycle rate), let us now develop a general expression for the summation of one of the j, k product terms in (10.1.2.2.2-11).

From the (10.1.2.2.2-11) definitions, we first write as in (10.1.2.2.2-8):

$$
\begin{aligned}
\Delta\underline{\alpha}_j &= \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_j - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_x}\right) \right] \\
\Delta\underline{\upsilon}_j &= \underline{u}_y\, a_{SF0_y} \frac{1}{\Omega} \left[\sin\left(\Omega\, t_j - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) \right] \\
\Delta\underline{\alpha}_k &= \underline{u}_x\, \theta_{0_x} \left[\sin\left(\Omega\, t_k - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{\theta_x}\right) \right] \\
\Delta\underline{\upsilon}_k &= \underline{u}_y\, a_{SF0_y} \frac{1}{\Omega} \left[\sin\left(\Omega\, t_k - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) \right]
\end{aligned}
\qquad (10.1.2.2.2\text{-}12)
$$

Combining Equations (10.1.2.2.2-12) into the (10.1.2.2.2-11) inner summation cross-product gives:

$$
\begin{aligned}
&\Delta\underline{\alpha}_j \times \Delta\underline{\upsilon}_k + \Delta\underline{\upsilon}_j \times \Delta\underline{\alpha}_k = \underline{u}_z\, \theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega} \Bigg\{ \left[\sin\left(\Omega\, t_j - \varphi_{\theta_x}\right) \right.\\
&\left. - \sin\left(\Omega\, t_{j-1} - \varphi_{\theta_x}\right) \right] \left[\sin\left(\Omega\, t_k - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) \right] \qquad (10.1.2.2.2\text{-}13)\\
&- \left[\sin\left(\Omega\, t_j - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) - \sin\left(\Omega\, t_{j-1} - \varphi_{a_{SF_y}} - \frac{\pi}{2}\right) \right] \left[\sin\left(\Omega\, t_k - \varphi_{\theta_x}\right) - \sin\left(\Omega\, t_{k-1} - \varphi_{\theta_x}\right) \right] \Bigg\}
\end{aligned}
$$

If we compare Equation (10.1.2.2.2-13) with the equivalent coning algorithm term $\Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k$ in Equation (10.1.1.2.2-12), we see that they are identical in form; (10.1.1.2.2-12) can be converted to (10.1.2.2.2-13) by substituting $\theta_{0_y} \rightarrow a_{SF0_y} \frac{1}{\Omega}$ and $\varphi_{\theta_y} \rightarrow \varphi_{a_{SF_y}} + \frac{\pi}{2}$. Then Equation (10.1.1.2.2-16) for $\Delta\underline{\alpha}_j \times \Delta\underline{\alpha}_k$ derived from (10.1.1.2.2-12) can be converted to the equivalent result needed for the (10.1.2.2.2-11) sculling algorithm using the same previous substitutions. Thus:

$$
\begin{aligned}
\Delta\underline{\alpha}_j \times \Delta\underline{\upsilon}_k + \Delta\underline{\upsilon}_j \times \Delta\underline{\alpha}_k &= \underline{u}_z\, \theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega} \sin\left(\varphi_{a_{SF_y}} + \frac{\pi}{2} - \varphi_{\theta_x}\right) \left[2 \sin p_{jk}\, \Omega\, T_l \right.\\
&\left. \quad - \sin\left(p_{jk} - 1\right) \Omega\, T_l - \sin\left(p_{jk} + 1\right) \Omega\, T_l \right] \qquad (10.1.2.2.2\text{-}14)\\
&= \underline{u}_z\, \theta_{0_x}\, a_{SF0_y} \frac{1}{\Omega} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \left[2 \sin p_{jk}\, \Omega\, T_l - \sin\left(p_{jk} - 1\right) \Omega\, T_l - \sin\left(p_{jk} + 1\right) \Omega\, T_l \right]
\end{aligned}
$$

where

p_{jk} = Integer representing the number of l cycles from cycle j at t_j to cycle k at t_k.

Equation (10.1.2.2.2-14) expresses the interesting result that for the Equation (10.1.2-2) and (10.1.2-3) angular-rate/linear-acceleration with $\Omega_x = \Omega_y = \Omega$, the sum of cross-products between integrated angular rate and linear acceleration increments separated by any integer number of sensor increment integration cycles, is constant.

Carrying the coning to sculling algorithm conversion process a step further, the Equation (10.1.1.2.2-19) summation term for the coning algorithm (which was derived from (10.1.1.2.2-12)) can be converted to the equivalent result for the sculling algorithm using the $\theta_{0_y} \rightarrow a_{SF0_y} \frac{1}{\Omega}$ and $\varphi_{\theta_y} \rightarrow \varphi_{a_{SF_y}} + \frac{\pi}{2}$ substitutions. Thus:

$$\sum_{l=(m-1)s+1}^{ms} \left(\Delta\underline{\alpha}_j \times \Delta\underline{\upsilon}_k + \Delta\underline{\upsilon}_j \times \Delta\underline{\alpha}_k \right) = \underline{u}_z \, \theta_{0_x} \, a_{SF0_y} \sin\left(\varphi_{a_{SF_y}} + \frac{\pi}{2} - \varphi_{\theta_x} \right) \left(2 \frac{\sin p_{jk} \, \Omega \, T_l}{\Omega \, T_l} - \frac{\sin\left(p_{jk} - 1\right) \Omega \, T_l}{\Omega \, T_l} - \frac{\sin\left(p_{jk} + 1\right) \Omega \, T_l}{\Omega \, T_l} \right) T_m \tag{10.1.2.2.2-15}$$

$$= \underline{u}_z \, \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x} \right) \left(2 \frac{\sin p_{jk} \, \Omega \, T_l}{\Omega \, T_l} - \frac{\sin\left(p_{jk} - 1\right) \Omega \, T_l}{\Omega \, T_l} - \frac{\sin\left(p_{jk} + 1\right) \Omega \, T_l}{\Omega \, T_l} \right) T_m$$

For the particular Section 7.2.2.2.2 sculling algorithm represented by (10.1.2.2.2-9), $p_{jk} = 1$ and Cj,k = 1/12 so with (10.1.2.2.2-15):

$$\sum_{l=(m-1)s+1}^{ms} \frac{1}{12} \left(\Delta\underline{\alpha}_{l-1} \times \Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1} \times \Delta\underline{\alpha}_l \right)$$

$$= \underline{u}_z \, \frac{1}{12} \, \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x} \right) \left(2 \frac{\sin \Omega \, T_l}{\Omega \, T_l} - \frac{\sin 2 \, \Omega \, T_l}{\Omega \, T_l} \right) T_m \tag{10.1.2.2.2-16}$$

$$= \underline{u}_z \, \frac{1}{6} \, \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x} \right) \frac{\sin \Omega \, T_l}{\Omega \, T_l} \left(1 - \cos \Omega \, T_l \right) T_m$$

Then, combining (10.1.2.2.2-10) with (10.1.2.2.2-16) in (10.1.2.2.2-9) and factoring 1/2 to the left, we obtain for the INS software sculling algorithm solution $\Delta\underline{v}_{Scul/Algo_m}$:

$$\Delta\underline{v}_{Scul/Algo_m} = \underline{u}_z \, \frac{1}{2} \, \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x} \right) \left[\left(\frac{\sin \Omega \, T_l}{\Omega \, T_l} - \frac{\sin \Omega \, T_m}{\Omega \, T_m} \right) + \frac{1}{3} \left(\frac{\sin \Omega \, T_l}{\Omega \, T_l} \left(1 - \cos \Omega \, T_l \right) \right) \right] T_m \tag{10.1.2.2.2-17}$$

$$= \underline{u}_z \, \frac{1}{2} \, \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x} \right) \left\{ \left[1 + \frac{1}{3} \left(1 - \cos \Omega \, T_l \right) \right] \left(\frac{\sin \Omega \, T_l}{\Omega \, T_l} \right) - \frac{\sin \Omega \, T_m}{\Omega \, T_m} \right\} T_m$$

Since $\Delta \underline{v}_{Scul/Algo_m}$ is constant in (10.1.2.2.2-17) and independent of m, its sum in (10.1.2.2.2-6) is easily evaluated as m times $\Delta \underline{v}_{Scul/Algo_m}$. Identifying the product of m with T_m as $t_m - t_0$, we then have:

$$\sum_{1}^{m} \Delta \underline{v}_{Scul/Algo_i} = \underline{u}_z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \left\{ \left[1 + \frac{1}{3} \left(1 - \cos \Omega T_l \right) \right] \left(\frac{\sin \Omega T_l}{\Omega T_l} \right) - \frac{\sin \Omega T_m}{\Omega T_m} \right\} (t_m - t_0) \tag{10.1.2.2.2-18}$$

Finally, we substitute (10.1.2.2.2-18) and $\sum_{1}^{m} \Delta \underline{v}_{Scul_i}$ from (10.1.2.2.1-10) in (10.1.2.2.2-6) to obtain for the total INS algorithm velocity error at computer cycle m:

$$\delta \underline{v}_{SF/Algo_m} = \underline{u}_z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \left\{ \left[1 + \frac{1}{3} \left(1 - \cos \Omega T_l \right) \right] \left(\frac{\sin \Omega T_l}{\Omega T_l} \right) - 1 \right\} (t_m - t_0) \tag{10.1.2.2.2-19}$$

We also substitute (10.1.2.2.2-17) for $\Delta \underline{v}_{Scul/Algo_m}$ with (10.1.2.2.1-9) for $\Delta \underline{v}_{Scul_m}$ into (10.1.2.2.2-7) which gives the INS sculling algorithm error for computer cycle m:

$$\delta \Delta \underline{v}_{Scul/Algo_m} = \underline{u}_z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \left\{ \left[1 + \frac{1}{3} \left(1 - \cos \Omega T_l \right) \right] \left(\frac{\sin \Omega T_l}{\Omega T_l} \right) - 1 \right\} T_m \tag{10.1.2.2.2-20}$$

Thus we see from (10.1.2.2.2-19) and (10.1.2.2.2-20) that the errors in the INS total velocity solution and sculling algorithm are proportional to the associated evaluation time interval, with the proportionality coefficient independent of time. As in Section 10.1.2.1 Equation (10.1.2.1-13) and Section 10.1.2.2.1 Equation (10.1.2.2.1-11), it is meaningful to identify the coefficients in (10.1.2.2.2-17), (10.1.2.2.2-19) and (10.1.2.2.2-20) as the sculling algorithm rate, the velocity algorithm rate error and the sculling algorithm rate error given by:

$$\Delta\dot{\underline{v}}_{Scul/Algo_m} = \underline{u}_z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \left\{ \left[1 + \frac{1}{3}\left(1 - \cos \Omega T_l\right) \right] \left(\frac{\sin \Omega T_l}{\Omega T_l} \right) - \frac{\sin \Omega T_m}{\Omega T_m} \right\} \tag{10.1.2.2.2-21}$$

$$\delta\dot{\underline{v}}_{SF/Algo_m} = \delta\Delta\dot{\underline{v}}_{Scul/Algo_m} = \underline{u}_z \frac{1}{2} \theta_{0_x} a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \left\{ \left[1 + \frac{1}{3}\left(1 - \cos \Omega T_l\right) \right] \left(\frac{\sin \Omega T_l}{\Omega T_l} \right) - 1 \right\} \tag{10.1.2.2.2-22}$$

where

$\Delta\dot{\underline{v}}_{Scul/Algo_m}$ = Constant rate of change in $\underline{v}_{SF/Algo_m}$ generated by the summing of $\Delta\underline{v}_{Scul/Algo_i}$'s in Equation (10.1.2.2.2-1).

$\delta\dot{\underline{v}}_{SF/Algo_m}$, $\delta\Delta\dot{\underline{v}}_{Scul/Algo_m}$ = Error build-up rate associated with INS software total velocity and the sculling algorithm contribution to total velocity.

10.1.3 POSITION RESPONSE UNDER LINEAR VIBRATION

Unlike the attitude and velocity response characteristics, the position computation response of inertial navigation systems under vibration can become significantly distorted due to "frequency folding" effects. Frequency folding is the apparent distortion one observes when sampling a sinusoidal wave form when the oscillation frequency is nearly an integer multiple of the sampling frequency. A classic example of frequency folding is the visual image generated from a strobe light illuminating a vibrating object. As the strobe light frequency is adjusted such that the vibration frequency is nearly an integer multiple of the strobe frequency, the illuminated object vibration frequency appears to slow down, becoming stationary when the ratio between the strobe and vibration frequencies is an exact integer.

In the case of a vibrating strapdown inertial navigation system, sampling the inertial sensor signals into the navigation computer produces a similar frequency folding effect. Thus, the attitude and velocity generated by integrating (summing) the angular rate sensor and accelerometer signals also contains frequency folding. If the angular rate sensor and accelerometer inputs represent integrated increments of angular rate and specific force acceleration (as assumed in the Chapters 7 and 19 (Section 19.1) integration algorithms), their sum in the navigation computer will generate the correct attitude/velocity solution at the sample instants, even though the apparent frequency will appear distorted due to the sample data folding effect. (The previous statement ignores the small, though generally not negligible coning/sculling effects present in the navigation computer generated attitude/velocity solutions under angular and linear vibrations whose accuracy is a function of the algorithm repetition rate. When the vibration effects are uncorrelated between axes (i.e., for zero coning and sculling), the attitude and velocity algorithm solutions presented in Chapters 7 and 19 (Section 19.1) would be exact at the computer cycle times, regardless of the algorithm update frequency). However, in the case of the position algorithm (i.e., digital integration of velocity), folding can produce a significant error in the position solution, even for uncorrelated between axis vibrations, because the integration is performed using folded velocity data. It is important to note that this same folding effect can also be present in the algorithm used to calculate the Chapter 6 initial alignment measurement ($\Delta \underline{R}_H^N$ in Section 6.1.2) which is also a velocity integration operation. The impact of folding effect algorithm error on initial heading alignment accuracy can be catastrophic in a vibration environment if not properly handled (See Sections 7.4 and 7.4.1 for further discussion). In general, potentially large position folding effect errors under linear vibration can only be eliminated by increasing the position update algorithm computation rate.

In the sub-sections to follow we will analyze the position computation frequency folding error in the presence of sinusoidal linear specific force acceleration vibration along a single axis, but with no angular motion. The acceleration vibration profile to be analyzed will be of the following general form:

$$\underline{a}_{SF}(t) = \underline{u}_{Vib}\, a_{SF_0} \sin\left(\Omega\, t - \varphi_{a_{SF}}\right) \qquad \underline{\omega}_{IB}(t) = 0 \qquad (10.1.3\text{-}1)$$

where

$\underline{\omega}_{IB}(t)$ = B Frame angular rate vector relative to inertial space that would be measured by the strapdown angular rate sensors.

$\underline{a}_{SF}(t)$ = B Frame specific force acceleration vector that would be measured by the strapdown accelerometers.

$\underline{u}_{Vib}$ = Linear vibration axis (assumed fixed in the B Frame).

Ω = Frequency of the $\underline{a}_{SF}(t)$ vibration.

a_{SF_0} = Amplitude of the $\underline{a}_{SF}(t)$ vibration.

$\varphi_{a_{SF}}$ = Phase angle associated with the $\underline{a}_{SF}(t)$ vibration.

10.1.3.1 POSITION MOTION CHARACTERISTICS

Let us analyze the effect of the Equation (10.1.3-1) angular rate and linear acceleration on position motion at time t relative to some arbitrary time t_0. To do this, it is convenient to define position as the integral of velocity in a non-rotating coordinate frame. The non-rotating frame we select is the B Frame at time t_0. Thus, we write:

$$\dot{\underline{R}}_{SF}(t) = \underline{v}_{SF}(t) \tag{10.1.3.1-1}$$

where

$\underline{R}_{SF}(t), \underline{v}_{SF}(t)$ = Position and velocity at time t in the time t_0 oriented B Frame due to (10.1.3-1) angular rate and linear vibration since time t_0.

We can use (10.1.2.1-3) for $\underline{v}_{SF}(t)$ in (10.1.3.1-1), which under the (10.1.3-1) zero assumed angular rate condition is from (10.1.2.1-1):

$$\dot{\underline{v}}_{SF}(t) = \underline{a}_{SF}(t) \tag{10.1.3.1-2}$$

The integral of (10.1.3.1-1) over a position computation update is the position change over the cycle:

$$\Delta\underline{R}_{SF_m} = \int_{t_{m-1}}^{t_m} \underline{v}_{SF}(t)\, dt \tag{10.1.3.1-3}$$

where

m = Position computation cycle index.

$\Delta\underline{R}_{SF_m}$ = Change in $\underline{R}_{SF}$ over cycle m.

The position at a particular computation cycle is the sum of $\Delta\underline{R}_{SF_m}$'s since t_0, up to the end of the particular computation cycle time:

$$\underline{R}_{SF_M} = \sum_{m=1}^{M} \Delta\underline{R}_{SF_m} \tag{10.1.3.1-4}$$

where

M = The value for m at a particular computation cycle.

$\underline{R}_{SF_M}$ = $\underline{R}_{SF}(t)$ at completion of cycle M (i.e., at time t_M).

Equation (10.1.3.1-2) can be integrated from time t_0. Assuming that $v_{SF}(t) = 0$ at time t_0 we get:

$$\underline{v}_{SF}(t) = \underline{v}_{SF_{m-1}} + \int_{t_{m-1}}^{t} \underline{a}_{SF}(\tau)\, d\tau \tag{10.1.3.1-5}$$

$$\underline{v}_{SF_{m-1}} = \int_{t_0}^{t_{m-1}} \underline{a}_{SF}(t)\, dt \tag{10.1.3.1-6}$$

where

$\underline{v}_{SF_{m-1}} = \underline{v}_{SF}(t)$ at time t_{m-1} (with $\underline{v}_{SF_0}$ assumed to be zero).

Substituting (10.1.3.1-5) in (10.1.3.1-3) yields for $\Delta\underline{R}_{SF_m}$:

$$\Delta\underline{R}_{SF_m} = \int_{t_{m-1}}^{t_m} \left(\underline{v}_{SF_{m-1}} + \int_{t_{m-1}}^{t} \underline{a}_{SF}(\tau)\, d\tau \right) dt = \underline{v}_{SF_{m-1}}\, T_m + \underline{S}_{\upsilon_m} \qquad (10.1.3.1\text{-}7)$$

in which we have defined as in (7.3.3-10):

$$\underline{S}_{\upsilon_m} \equiv \int_{t_{m-1}}^{t_m} \left(\int_{t_{m-1}}^{t} \underline{a}_{SF}(\tau)\, d\tau \right) dt \qquad (10.1.3.1\text{-}8)$$

where

$\underline{S}_{\upsilon_m}$ = Double integral of $a_{SF}(t)$ over cycle m.

T_m = The time interval for each m cycle.

Substituting (10.1.3.1-7) in (10.1.3.1-4) then gives:

$$\underline{R}_{SF_M} = \sum_{m=1}^{M} \underline{v}_{SF_{m-1}}\, T_m + \sum_{m=1}^{M} \underline{S}_{\upsilon_m} \qquad (10.1.3.1\text{-}9)$$

Equation (10.1.3.1-9) with (10.1.3.1-8) and (10.1.3.1-6) represents the actual position response characteristic of the sensor assembly under zero angular rate.

10.1.3.2 POSITION ALGORITHM RESPONSE

The software in a strapdown INS processes a digital velocity integration algorithm to calculate position. Chapters 7 and 19 (Section 19.1) describe such position algorithms as a general inertial navigation computational operation (Sections 7.3 and 19.1.5). Similar algorithms would be used to calculate the Chapter 6 initial alignment measurement ($\Delta\underline{R}_H^N$ in Section 6.1.2) which is also a velocity integration operation. Equation (10.1.3.1-9) with (10.1.3.1-8) and (10.1.3.1-6) represents an exact integration algorithm that would generate the exact position solution. In an INS, the equivalent to these equations would be approximated in the INS computer. The following subsections analyze the response of Equation (10.1.3.1-9) with (10.1.3.1-8) and (10.1.3.1-6) under the hypothesized (10.1.3-1) sinusoidal linear vibration for an exact algorithmic implementation and for an approximate form typified in the INS software.

10.1.3.2.1 Exact Position Algorithm Response

Under the (10.1.3-1) sinusoidal linear vibration profile, the inner term in (10.1.3.1-8) can be analytically integrated as follows:

$$\int_{t_{m-1}}^{t} \underline{a}_{SF}(\tau)\, d\tau = - \underline{u}_{Vib} \frac{1}{\Omega} a_{SF_0} \left[\cos\left(\Omega\, t - \varphi_{a_{SF}}\right) - \cos\left(\Omega\, t_{m-1} - \varphi_{a_{SF}}\right) \right] \qquad (10.1.3.2.1\text{-}1)$$

Substituting (10.1.3.2.1-1) in (10.1.3.1-8) gives:

$$\begin{aligned} \underline{S}_{\upsilon_m} = - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \Big\{ & \left[\sin\left(\Omega\, t_m - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_{m-1} - \varphi_{a_{SF}}\right) \right] \\ & - T_m\, \Omega \cos\left(\Omega\, t_{m-1} - \varphi_{a_{SF}}\right) \Big\} \end{aligned} \qquad (10.1.3.2.1\text{-}2)$$

Applying (10.1.3.2.1-2) to the summation term in (10.1.3.1-9) obtains:

$$\begin{aligned} \sum_{m=1}^{M} \underline{S}_{\upsilon_m} = - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \Bigg[& \sin\left(\Omega\, t_M - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \\ & - \sum_{m=1}^{M} T_m\, \Omega \cos\left(\Omega\, t_{m-1} - \varphi_{a_{SF}}\right) \Bigg] \end{aligned} \qquad (10.1.3.2.1\text{-}3)$$

Finally, we substitute (10.1.3.2.1-3) into (10.1.3.1-9) to find with (10.1.3.1-6):

$$\begin{aligned} \underline{R}_{SF_M} = & - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \left[\sin\left(\Omega\, t_M - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \right] \\ & + \sum_{m=1}^{M} \left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib} \frac{1}{\Omega} a_{SF_0} \cos\left(\Omega\, t_{m-1} - \varphi_{a_{SF}}\right) \right] T_m \end{aligned} \qquad (10.1.3.2.1\text{-}4)$$

$$\underline{v}_{SF_{m-1}} = \int_{t_0}^{t_{m-1}} \underline{a}_{SF}(t)\, dt \qquad (10.1.3.2.1\text{-}5)$$

Equation (10.1.3.2.1-4) with (10.1.3.2.1-5) represents the response of the exact position updating algorithm to the (10.1.3-1) vibration profile.

10.1.3.2.2 INS Position Algorithm Response

In an INS computer, the position updating algorithm equivalent to Equation (10.1.3.1-9) with (10.1.3.1-8) and (10.1.3.1-6) can be represented as:

$$\underline{R}_{SF/Algo_M} = \sum_{m=1}^{M} \underline{v}_{SF/Algo_{m-1}} T_m + \sum_{m=1}^{M} \underline{S}_{\upsilon/Algo_m} \tag{10.1.3.2.2-1}$$

$$\underline{v}_{SF/Algo_{m-1}} = \sum_{i=1}^{m-1} \Delta\underline{v}_{SF/Algo_i} \tag{10.1.3.2.2-2}$$

$$\underline{S}_{\upsilon/Algo_m} = \sum_{l=(m-1)r+1}^{m\,r} \Delta\underline{S}_{\upsilon/Algo_l} \tag{10.1.3.2.2-3}$$

where

Algo = Designation for the INS algorithm calculated value of the associated parameter.

i = Dummy m cycle index.

l = High speed computation cycle index.

r = Number of l cycles in an m cycle.

$\Delta\underline{v}_{SF/Algo_i}$ = The change in $\underline{v}_{SF/Algo}$ over computer cycle m = i.

$\Delta\underline{S}_{\upsilon/Algo_l}$ = The change in $\underline{S}_{\upsilon/Algo}$ over computer cycle l.

For zero angular rate, we see from (10.1.2.2.2-1), (10.1.2.2.2-3) and $d\underline{\upsilon} = \underline{a}_{SF}(t)\,dt$ (using the definition following (7.2.2.2.2-15)) that $\Delta\underline{v}_{SF/Algo_i}$ is given by:

$$\Delta\underline{v}_{SF/Algo_i} = \sum_{l=(i-1)r+1}^{i\,r} \Delta\underline{\upsilon}_l \qquad \Delta\underline{\upsilon}_l = \int_{t_{l-1}}^{t_l} \underline{a}_{SF}(t)\,dt \tag{10.1.3.2.2-4}$$

where

$\Delta\underline{\upsilon}_l$ = Integrated specific force over an l cycle, the form of the input assumed from the strapdown accelerometers.

Substituting (10.1.3.2.2-4) in (10.1.3.2.2-2) finds the obvious result:

$$\underline{v}_{SF/Algo_{m-1}} = \sum_{i=1}^{m-1} \sum_{l=(i-1)r+1}^{i\,r} \Delta\underline{\upsilon}_l = \sum_{l=1}^{(m-1)r} \Delta\underline{\upsilon}_l = \int_{t_0}^{t_{m-1}} \underline{a}_{SF}(t)\,dt \tag{10.1.3.2.2-5}$$

Thus, by comparison with (10.1.3.2.1-5):

$$\underline{v}_{SF/Algo_{m-1}} = \underline{v}_{SF_{m-1}} \tag{10.1.3.2.2-6}$$

i.e., the velocity computation algorithm is error free in the absence of angular rate.

For the $\Delta\underline{S}_{\upsilon/Algo_l}$ term in (10.1.3.2.2-3) we can write from (7.3.3.2-19) and (7.2.2.2.2-14) for Chapter 7 high resolution position algorithm (and for the Chapter 19 (Section 19.1), Equations (19.1.11-1) position translation vector algorithm):

$$\Delta\underline{S}_{\upsilon/Algo_l} = \underline{\upsilon}_{l-1} T_l + \frac{T_l}{12}\left(5\,\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right) \qquad \underline{\upsilon}_l = \underline{\upsilon}_{l-1} + \Delta\underline{\upsilon}_l \qquad \underline{\upsilon}_{(m-1)r} = 0 \tag{10.1.3.2.2-7}$$

where

T_l = The time interval for each l cycle.

From (10.1.3.2.2-7) we see that:

$$\Delta\underline{\upsilon}_l = \underline{\upsilon}_l - \underline{\upsilon}_{l-1} \tag{10.1.3.2.2-8}$$

with which the $\frac{T_l}{12}\left(5\,\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right)$ term in (10.1.3.2.2-7) can be expressed as:

$$\begin{aligned}\frac{T_l}{12}\left(5\,\Delta\underline{\upsilon}_l + \Delta\underline{\upsilon}_{l-1}\right) &= \frac{T_l}{12}\left[6\,\Delta\underline{\upsilon}_l - \left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\right] = \frac{1}{2}\,\Delta\underline{\upsilon}_l\, T_l - \frac{T_l}{12}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) \\ &= \frac{1}{2}\left(\underline{\upsilon}_l - \underline{\upsilon}_{l-1}\right) T_l - \frac{T_l}{12}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right)\end{aligned} \tag{10.1.3.2.2-9}$$

Substituting (10.1.3.2.2-9) in (10.1.3.2.2-7) and the result in (10.1.3.2.2-3) for $\underline{S}_{\upsilon/Algo_m}$, rewriting the (10.1.3.2.2-7) $\underline{\upsilon}_l$ term in summation form, and including (10.1.3.2.2-4) for $\Delta\underline{\upsilon}_l$ yields the equivalent expressions:

$$\begin{gathered}\underline{S}_{\upsilon/Algo_m} = \sum_{l=(m-1)r+1}^{m\,r} \frac{1}{2}\left(\underline{\upsilon}_l + \underline{\upsilon}_{l-1}\right) T_l - \sum_{l=(m-1)r+1}^{m\,r} \frac{1}{12}\left(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\right) T_l \\ \underline{\upsilon}_l = \sum_{j=(m-1)r+1}^{l} \Delta\underline{\upsilon}_j \qquad \underline{\upsilon}_{(m-1)r} = 0 \qquad \Delta\underline{\upsilon}_l = \int_{t_{l-1}}^{t_l} \underline{a}_{SF}(t)\, dt\end{gathered} \tag{10.1.3.2.2-10}$$

where

j = Dummy index for l.

Equations (10.1.3.2.2-10) are in the form we need for evaluation of $\underline{S}_{\upsilon/\mathrm{Algo}_m}$ based on the particular $\underline{a}_{SF}(t)$ vibration profile being analyzed. Substituting $\underline{a}_{SF}(t)$ from (10.1.3-1) in the (10.1.3.2.2-10) $\Delta\underline{\upsilon}_l$ and $\underline{\upsilon}_l$ expressions yields:

$$
\begin{aligned}
\Delta\underline{\upsilon}_l &= -\underline{u}_{Vib}\,\frac{1}{\Omega}\,a_{SF_0}\Big(\cos(\Omega\, t_l - \varphi_{a_{SF}}) - \cos(\Omega\, t_{l-1} - \varphi_{a_{SF}})\Big)\\
\Delta\underline{\upsilon}_{l-1} &= -\underline{u}_{Vib}\,\frac{1}{\Omega}\,a_{SF_0}\Big(\cos(\Omega\, t_{l-1} - \varphi_{a_{SF}}) - \cos(\Omega\, t_{l-2} - \varphi_{a_{SF}})\Big)\\
\underline{\upsilon}_l &= -\underline{u}_{Vib}\,\frac{1}{\Omega}\,a_{SF_0}\Big(\cos(\Omega\, t_l - \varphi_{a_{SF}}) - \cos(\Omega\, t_{(m-1)r} - \varphi_{a_{SF}})\Big)\\
\underline{\upsilon}_{l-1} &= -\underline{u}_{Vib}\,\frac{1}{\Omega}\,a_{SF_0}\Big(\cos(\Omega\, t_{l-1} - \varphi_{a_{SF}}) - \cos(\Omega\, t_{(m-1)r} - \varphi_{a_{SF}})\Big)\\
\underline{\upsilon}_{(m-1)r} &= 0
\end{aligned}
\qquad (10.1.3.2.2\text{-}11)
$$

Using the $\Delta\underline{\upsilon}_l$ terms from (10.1.3.2.2-11), $\sum_{l\,=\,(m-1)r+1}^{mr} \frac{1}{12}\Big(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\Big)\,T_l$ in (10.1.3.2.2-10) becomes:

$$
\sum_{l\,=\,(m-1)r+1}^{mr} \frac{1}{12}\Big(\Delta\underline{\upsilon}_l - \Delta\underline{\upsilon}_{l-1}\Big)\,T_l = -\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\,\frac{1}{12}\Big[\cos(\Omega\, t_{mr} - \varphi_{a_{SF}}) - \cos(\Omega\, t_{mr-1} - \varphi_{a_{SF}}) - \cos(\Omega\, t_{(m-1)r} - \varphi_{a_{SF}}) + \cos(\Omega\, t_{(m-1)r-1} - \varphi_{a_{SF}})\Big]\,\Omega\, T_l \qquad (10.1.3.2.2\text{-}12)
$$

The $\sum_{l\,=\,(m-1)r+1}^{mr} \frac{1}{2}\Big(\underline{\upsilon}_l + \underline{\upsilon}_{l-1}\Big)\,T_l$ term in (10.1.3.2.2-10) with $\underline{\upsilon}_{(m-1)r} = 0$ (from (10.1.3.2.2-10)) is first expanded as:

$$
\begin{aligned}
\sum_{l\,=\,(m-1)r+1}^{mr} \frac{1}{2}\Big(\underline{\upsilon}_l + \underline{\upsilon}_{l-1}\Big)\,T_l &= \frac{1}{2}\Big(\underline{\upsilon}_{(m-1)r+1} + \underline{\upsilon}_{(m-1)r}\Big)\,T_l + \sum_{l\,=\,(m-1)r+2}^{mr} \frac{1}{2}\Big(\underline{\upsilon}_l + \underline{\upsilon}_{l-1}\Big)\,T_l\\
&= \frac{1}{2}\,\underline{\upsilon}_{(m-1)r+1}\,T_l + \sum_{l\,=\,(m-1)r+2}^{mr} \frac{1}{2}\Big(\underline{\upsilon}_l + \underline{\upsilon}_{l-1}\Big)\,T_l
\end{aligned}
\qquad (10.1.3.2.2\text{-}13)
$$

We then substitute from (10.1.3.2.2-11) for the $\underline{\upsilon}_l$'s:

$$\underline{\upsilon}_l + \underline{\upsilon}_{l-1} = -\,\underline{u}_{Vib}\,\frac{1}{\Omega}\,a_{SF_0}\Big[\cos(\Omega\,t_l - \varphi_{a_{SF}}) + \cos(\Omega\,t_{l-1} - \varphi_{a_{SF}}) - 2\cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Big] \tag{10.1.3.2.2-14}$$

$$\underline{\upsilon}_{(m-1)r+1} = -\,\underline{u}_{Vib}\,\frac{1}{\Omega}\,a_{SF_0}\Big[\cos(\Omega\,t_{(m-1)r+1} - \varphi_{a_{SF}}) - \cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Big]$$

Applying (10.1.3.2.2-14) in (10.1.3.2.2-13) yields:

$$\begin{aligned}
&\sum_{l=(m-1)r+1}^{mr} \frac{1}{2}\left(\underline{\upsilon}_l + \underline{\upsilon}_{l-1}\right) T_l \\
&= -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\Bigg\{\frac{1}{2}\Big[\cos(\Omega\,t_{(m-1)r+1} - \varphi_{a_{SF}}) - \cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Big]\,\Omega\,T_l \\
&\qquad + \sum_{l=(m-1)r+2}^{mr} \frac{1}{2}\Big[\cos(\Omega\,t_l - \varphi_{a_{SF}}) + \cos(\Omega\,t_{l-1} - \varphi_{a_{SF}})\Big]\,\Omega\,T_l \\
&\qquad - \Omega\,(r-1)\;T_l\cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Bigg\} \\
&= -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\Bigg\{\frac{1}{2}\Big[\cos(\Omega\,t_{(m-1)r+1} - \varphi_{a_{SF}}) - \cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Big]\,\Omega\,T_l \\
&\qquad + \sum_{l=(m-1)r+1}^{mr} \frac{1}{2}\Big[\cos(\Omega\,t_l - \varphi_{a_{SF}}) + \cos(\Omega\,t_{l-1} - \varphi_{a_{SF}})\Big]\,\Omega\,T_l \\
&\qquad - \frac{1}{2}\Big[\cos(\Omega\,t_{(m-1)r+1} - \varphi_{a_{SF}}) + \cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Big]\,\Omega\,T_l \\
&\qquad - \Omega\,(r-1)\;T_l\cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Bigg\} \\
&= -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\Bigg\{\sum_{l=(m-1)r+1}^{mr} \frac{1}{2}\Big[\cos(\Omega\,t_l - \varphi_{a_{SF}}) + \cos(\Omega\,t_{l-1} - \varphi_{a_{SF}})\Big]\,\Omega\,T_l \\
&\qquad - \cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\,\Omega\,T_l - \Omega\,(r-1)\;T_l\cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Bigg\} \\
&= -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\Bigg\{\sum_{l=(m-1)r+1}^{mr} \frac{1}{2}\Big[\cos(\Omega\,t_l - \varphi_{a_{SF}}) + \cos(\Omega\,t_{l-1} - \varphi_{a_{SF}})\Big]\,\Omega\,T_l \\
&\qquad - \Omega\,r\,T_l\cos(\Omega\,t_{(m-1)r} - \varphi_{a_{SF}})\Bigg\}
\end{aligned} \tag{10.1.3.2.2-15}$$

Based on the definitions for m, r, and T_l, we can write for the time parameters in (10.1.3.2.2-15) and (10.1.3.2.2-12):

$$
\begin{aligned}
t_l &= l\, T_l + t_0 & t_{l-1} &= (l\text{-}1)\; T_l + t_0 \\
t_{mr} &= m\, r\, T_l + t_0 & t_{mr-1} &= (mr\text{-}1)\; T_l + t_0 \qquad\qquad (10.1.3.2.2\text{-}16)\\
t_{(m-1)r} &= (m\text{-}1)\, r\, T_l + t_0 & t_{(m-1)r-1} &= \big((m\text{-}1)\, r\text{-}1\big)\, T_l + t_0
\end{aligned}
$$

Continuing development is expedited by defining:

$$\beta \equiv \Omega\, T_l \qquad \alpha \equiv \Omega\, T_m \qquad \phi \equiv \Omega\, t_0 - \varphi_{a_{SF}} \qquad (10.1.3.2.2\text{-}17)$$

We also know from the definition of r and T_l that

$$T_m = r\, T_l \qquad (10.1.3.2.2\text{-}18)$$

so α in (10.1.3.2.2-17) is equivalently:

$$\alpha = \Omega\, r\, T_l \qquad (10.1.3.2.2\text{-}19)$$

Using (10.1.3.2.2-16), (10.1.3.2.2-17) and (10.1.3.2.2-19), the trigonometric arguments in (10.1.3.2.2-15) and (10.1.3.2.2-12) become:

$$
\begin{aligned}
\Omega\, t_l - \varphi_{a_{SF}} &= l\, \beta + \phi & \Omega\, t_{l-1} - \varphi_{a_{SF}} &= (l\text{-}1)\; \beta + \phi \\
\Omega\, t_{mr} - \varphi_{a_{SF}} &= m\, \alpha + \phi & \Omega\, t_{mr-1} - \varphi_{a_{SF}} &= m\, \alpha + \phi - \beta \qquad\qquad (10.1.3.2.2\text{-}20)\\
\Omega\, t_{(m-1)r} - \varphi_{a_{SF}} &= (m\text{-}1)\; \alpha + \phi & \Omega\, t_{(m-1)r-1} - \varphi_{a_{SF}} &= (m\text{-}1)\; \alpha + \phi - \beta
\end{aligned}
$$

Substituting (10.1.3.2.2-20) in (10.1.3.2.2-15) and (10.1.3.2.2-12), and the result with (10.1.3.2.2-19) in (10.1.3.2.2-10), then finds for $\underline{S}_{\upsilon/Algo_m}$:

$$
\begin{aligned}
\underline{S}_{\upsilon/Algo_m} = &- \underline{u}_{Vib}\, \frac{1}{\Omega^2}\, a_{SF_0} \left\langle \left\{ \sum_{l\, =\, (m-1)r+1}^{mr} \frac{1}{2} \left[\cos\,(l\, \beta + \phi) + \cos\big((l\text{-}1)\, \beta + \phi\big)\right] \beta \right\} \right. \\
&- \frac{1}{12}\, \beta \Big[\cos\,(m\;\alpha + \phi) - \cos\,\big((m\text{-}1)\;\alpha + \phi\big) - \cos\,(m\;\alpha + \phi - \beta) \qquad (10.1.3.2.2\text{-}21)\\
&\left. + \cos\,\big((m\text{-}1)\;\alpha + \phi - \beta\big)\Big] - \alpha\, \cos\,\big((m\text{-}1)\;\alpha + \phi\big) \right\rangle
\end{aligned}
$$

The $\frac{1}{12}\, \beta$ coefficient in (10.1.3.2.2-21) is equivalently:

$$\cos(m\,\alpha + \phi) - \cos\big((m\text{-}1)\,\alpha + \phi\big) - \cos(m\,\alpha + \phi - \beta) + \cos\big((m\text{-}1)\,\alpha + \phi - \beta\big)$$
$$= \cos(m\,\alpha + \phi) - \cos\big((m\text{-}1)\,\alpha + \phi\big) - \cos(m\,\alpha + \phi)\cos\beta - \sin(m\,\alpha + \phi)\sin\beta$$
$$+ \cos\big((m\text{-}1)\,\alpha + \phi\big)\cos\beta + \sin\big((m\text{-}1)\,\alpha + \phi\big)\sin\beta$$
$$= \Big[\cos(m\,\alpha + \phi) - \cos\big((m\text{-}1)\,\alpha + \phi\big)\Big](1 - \cos\beta) \qquad (10.1.3.2.2\text{-}22)$$
$$- \Big[\sin(m\,\alpha + \phi) - \sin\big((m\text{-}1)\,\alpha + \phi\big)\Big]\sin\beta$$

Applying (10.1.3.2.2-22) in (10.1.3.2.2-21) and summing the result from m = 1 to M finds an expression for $\sum_{m=1}^{M} \underline{S}_{\upsilon/Algo_m}$ in the (10.1.3.2.2-1) $\underline{R}_{SF/Algo_M}$ equation:

$$\sum_{m=1}^{M} \underline{S}_{\upsilon/Algo_m} = -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\Big\langle \beta \Big\{ \sum_{l=1}^{Mr} \frac{1}{2}\Big[\cos(l\,\beta + \phi) + \cos\big((l\text{-}1)\,\beta + \phi\big)\Big] \Big\}$$
$$- \frac{1}{12}\,\beta\Big[\big(\cos(M\,\alpha + \phi) - \cos\phi\big)(1 - \cos\beta) - \big(\sin(M\,\alpha + \phi) - \sin\phi\big)\sin\beta\Big] \qquad (10.1.3.2.2\text{-}23)$$
$$- \sum_{m=1}^{M} \alpha\cos\big((m\text{-}1)\,\alpha + \phi\big)\Big\rangle$$

We now substitute (10.1.3.2.2-23) and (10.1.3.2.2-6) in (10.1.3.2.2-1) to obtain an equation for the algorithm computed position vector:

$$\underline{R}_{SF/Algo_M} = -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\Big\langle \beta \Big\{ \sum_{l=1}^{Mr} \frac{1}{2}\Big[\cos(l\,\beta + \phi) + \cos\big((l\text{-}1)\,\beta + \phi\big)\Big] \Big\}$$
$$- \frac{1}{12}\,\beta\Big[\big(\cos(M\,\alpha + \phi) - \cos\phi\big)(1 - \cos\beta) - \big(\sin(M\,\alpha + \phi) - \sin\phi\big)\sin\beta\Big] \qquad (10.1.3.2.2\text{-}24)$$
$$- \sum_{m=1}^{M} \alpha\cos\big((m\text{-}1)\,\alpha + \phi\big)\Big\rangle + \sum_{m=1}^{M} \underline{v}_{SF_{m-1}}\,T_m$$

To complete the development we will need an equation for the leading (10.1.3.2.2-24) summation term. This is derived from basic trigonometric identities as follows. First we write in general:

$$\cos(\rho + \gamma) + \cos(\rho - \gamma) = 2\cos\gamma\cos\rho$$
$$\rho - \gamma = \rho + \gamma - 2\,\gamma \qquad (10.1.3.2.2\text{-}25)$$

where

ρ, γ = Arbitrary parameters.

Then we define:

$$\mu \ \equiv \ \rho + \gamma \tag{10.1.3.2.2-26}$$

where

μ = Another arbitrary parameter.

so that:

$$\rho \ = \ \mu - \gamma \tag{10.1.3.2.2-27}$$

Substituting (10.1.3.2.2-27) in (10.1.3.2.2-25) finds with rearrangement:

$$\frac{1}{2}\cos\,\mu \ = \ \cos\,\gamma\cos\,(\,\mu - \gamma) - \frac{1}{2}\cos\,(\,\mu - 2\,\gamma) \tag{10.1.3.2.2-28}$$

or upon expansion:

$$\frac{1}{2}\Big(\cos\,\mu + \cos\,(\,\mu - \ \gamma)\Big) \ = \ \cos\,\gamma\cos\,(\,\mu - \gamma) + \frac{1}{2}\Big(\cos\,(\,\mu - \ \gamma) - \cos\,(\,\mu - 2\,\gamma)\Big) \tag{10.1.3.2.2-29}$$

We now set the μ and γ parameters to particular parameters in Equation (10.1.3.2.2-24):

$$\mu \ = \ l\,\beta + \phi \qquad\qquad \gamma \ = \ \beta \tag{10.1.3.2.2-30}$$

and note from (10.1.3.2.2-30) that:

$$\mu - \gamma \ = \ (l\text{-}1)\ \beta + \phi \qquad\qquad \mu - 2\,\gamma \ = \ (l\text{-}2)\ \beta + \phi \tag{10.1.3.2.2-31}$$

Applying (10.1.3.2.2-30) and (10.1.3.2.2-31) in (10.1.3.2.2-29) obtains for the argument in the leading (10.1.3.2.2-24) summation term:

$$\begin{aligned}&\frac{1}{2}\Big[\cos\ (l\,\beta + \phi) + \cos\,\Big((l\text{-}1)\ \beta + \phi\Big)\Big] \\ &\quad = \ \cos\,\beta\,\cos\,\Big((l\text{-}1)\ \beta + \phi\Big) + \frac{1}{2}\Big(\cos\,\Big((l\text{-}1)\ \beta + \phi\Big) - \cos\,\Big((l\text{-}2)\ \beta + \phi\Big)\Big)\end{aligned} \tag{10.1.3.2.2-32}$$

The term multiplying $\cos\,\beta$ in (10.1.3.2.2-32) can be expanded as:

$$\begin{aligned}&\cos\,\Big((l\text{-}1)\ \beta + \phi\Big) \\ &\quad = \ \frac{1}{2}\Big[\cos\,\Big((l\text{-}1)\ \beta + \phi\Big) + \cos\,(\,l\,\beta + \phi) - \cos\,(\,l\,\beta + \phi) + \cos\,\Big((l\text{-}1)\ \beta + \phi\Big)\Big] \\ &\quad = \ \frac{1}{2}\Big[\cos\,(\,l\,\beta + \phi) + \cos\,\Big((l\text{-}1)\ \beta + \phi\Big)\Big] - \frac{1}{2}\Big[\cos\,(\,l\,\beta + \phi) - \cos\,\Big((l\text{-}1)\ \beta + \phi\Big)\Big]\end{aligned} \tag{10.1.3.2.2-33}$$

Substituting (10.1.3.2.2-33) in (10.1.3.2.2-32) obtains:

$$
\begin{aligned}
&\frac{1}{2}\left[\cos\,(l\,\beta+\phi)+\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right] \\
&= \cos\,\beta\left\{\frac{1}{2}\left[\cos\,(l\,\beta+\phi)+\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right]-\frac{1}{2}\left[\cos\,(l\,\beta+\phi)-\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right]\right\} \\
&\qquad\qquad + \frac{1}{2}\left[\cos\,\big((l\text{-}1)\;\beta+\phi\big)-\cos\,\big((l\text{-}2)\;\beta+\phi\big)\right] \qquad\qquad (10.1.3.2.2\text{-}34) \\
&= \cos\,\beta\,\frac{1}{2}\left[\cos\,(l\,\beta+\phi)+\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right]-\frac{1}{2}\cos\,\beta\left[\cos\,(l\,\beta+\phi)-\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right] \\
&\quad + \frac{1}{2}\left[\cos\,(l\,\beta+\phi)-\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right]\cos\,\beta+\frac{1}{2}\left[\sin\,(l\,\beta+\phi)-\sin\,\big((l\text{-}1)\;\beta+\phi\big)\right]\sin\,\beta \\
&= \cos\,\beta\,\frac{1}{2}\left[\cos\,(l\,\beta+\phi)+\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right]+\frac{1}{2}\sin\,\beta\left[\sin\,(l\,\beta+\phi)-\sin\,\big((l\text{-}1)\;\beta+\phi\big)\right]
\end{aligned}
$$

or upon rearrangement:

$$
\begin{aligned}
&\frac{1}{2}\left[\cos\,(l\,\beta+\phi)+\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right] \\
&\qquad = \frac{\sin\,\beta}{2\,(1-\cos\,\beta)}\left[\sin\,(l\,\beta+\phi)-\sin\,\big((l\text{-}1)\;\beta+\phi\big)\right]
\end{aligned}
\qquad (10.1.3.2.2\text{-}35)
$$

Equation (10.1.3.2.2-35) is in the form we seek for application in the leading (10.1.3.2.2-24) summation term. The result is:

$$
\begin{aligned}
&\sum_{l=1}^{Mr}\frac{1}{2}\left[\cos\,(l\,\beta+\phi)+\cos\,\big((l\text{-}1)\;\beta+\phi\big)\right] \\
&\qquad = \frac{\sin\,\beta}{2\,(1-\cos\,\beta)}\left(\sin\,(M\,r\;\beta+\phi)-\sin\;\phi\right)
\end{aligned}
\qquad (10.1.3.2.2\text{-}36)
$$

Substituting (10.1.3.2.2-36) in (10.1.3.2.2-24) then yields for $\underline{R}_{SF/Algo_M}$:

$$
\begin{aligned}
\underline{R}_{SF/Algo_M} = -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\Bigg\langle &\frac{\beta\sin\beta}{2\,(1-\cos\,\beta)}\left(\sin\,(M\,r\,\beta+\phi)-\sin\phi\right) \\
-\frac{1}{12}\,\beta\Big[&\left(\cos\,(M\;\alpha+\phi)-\cos\;\phi\right)(\,1-\cos\;\beta)-\left(\sin\,(M\;\alpha+\phi)-\sin\;\phi\right)\sin\,\beta\Big] \qquad (10.1.3.2.2\text{-}37) \\
&-\sum_{m=1}^{M}\alpha\cos\,\big((m\text{-}1)\;\alpha+\phi\big)\Bigg\rangle+\sum_{m=1}^{M}\underline{v}_{SF_{m-1}}\,T_m
\end{aligned}
$$

The final step is to convert the α, β, ϕ parameters in (10.1.3.2.2-37). This is easily achieved by noting from the T_m, M and m definitions that:

$$M\,T_m = t_M - t_0 \qquad (m-1)\,T_m = t_{m-1} - t_0 \tag{10.1.3.2.2-38}$$

and from (10.1.3.2.2-17) and (10.1.3.2.2-19):

$$\alpha = r\,\beta \tag{10.1.3.2.2-39}$$

where

t_M = Time t at computer cycle m = M.

Using (10.1.3.2.2-38) - (10.1.3.2.2-39) and (10.1.3.2.2-17) we see then that:

$$\begin{aligned}
M\,r\,\beta + \phi &= M\,\alpha + \phi = M\,\Omega\,T_m + \phi = \Omega\,(t_M - t_0) + \Omega\,t_0 - \varphi_{a_{SF}} \\
&= \Omega\,t_M - \varphi_{a_{SF}} \\
(m-1)\,\alpha + \phi &= (m-1)\,\Omega\,T_m + \phi = \Omega\,(t_{m-1} - t_0) + \Omega\,t_0 - \varphi_{a_{SF}} \\
&= \Omega\,t_{m-1} - \varphi_{a_{SF}}
\end{aligned} \tag{10.1.3.2.2-40}$$

Substituting (10.1.3.2.2-40) with α and β from (10.1.3.2.2-17) in (10.1.3.2.2-37) obtains the final result:

$$\begin{aligned}
\underline{R}_{SF/Algo_M} = &- \underline{u}_{Vib}\,\frac{1}{\Omega^2}\,a_{SF_0}\left\{\left(\frac{\Omega\,T_l \sin \Omega\,T_l}{2\left(1 - \cos \Omega\,T_l\right)} + \frac{1}{12}\,\Omega\,T_l \sin \Omega\,T_l\right)\Big[\sin\left(\Omega\,t_M - \varphi_{a_{SF}}\right)\right. \\
&\left. - \sin\left(\Omega\,t_0 - \varphi_{a_{SF}}\right)\Big] - \frac{1}{12}\,\Omega\,T_l\Big[\cos\left(\Omega\,t_M - \varphi_{a_{SF}}\right) - \cos\left(\Omega\,t_0 - \varphi_{a_{SF}}\right)\Big]\left(1 - \cos\,\Omega\,T_l\right)\right\} \\
&+ \sum_{m=1}^{M}\left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib}\,\frac{1}{\Omega}\,a_{SF_0} \cos\left(\Omega\,t_{m-1} - \varphi_{a_{SF}}\right)\right] T_m
\end{aligned} \tag{10.1.3.2.2-41}$$

Equation (10.1.3.2.2-41) represents the INS computer algorithm determined position response to the (10.1.3-1) input vibration, assuming that the position updating algorithm is of the Chapter 7 high resolution (or the Chapter 19 (Section 19.1), Equations (19.1.11-1) position translation vector algorithm) form.

10.1.3.2.3 Folding Effects In The Position Algorithms

Sections 10.1.3.2.1 and 10.1.3.2.2 developed analytical expressions for the INS position response to the (10.1.3-1) vibration input for exact and INS position updating algorithm implementations. The results provided by Equations (10.1.3.2.1-4), (10.1.3.2.1-5) and (10.1.3.2.2-41) are repeated below in slightly revised format:

$$\underline{R}_{SF/Algo_M} = - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \left\{ \left(\frac{\Omega T_l \sin \Omega T_l}{2 \left(1 - \cos \Omega T_l\right)} + \frac{1}{12} \Omega T_l \sin \Omega T_l \right) \left[\sin \left(\Omega (t_M - t_0) + \Omega t_0 - \varphi_{a_{SF}} \right) - \sin \left(\Omega t_0 - \varphi_{a_{SF}} \right) \right] \right.$$
$$\left. - \frac{1}{12} \Omega T_l \left[\cos \left(\Omega (t_M - t_0) + \Omega t_0 - \varphi_{a_{SF}} \right) - \cos \left(\Omega t_0 - \varphi_{a_{SF}} \right) \right] \left(1 - \cos \Omega T_l\right) \right\}$$
$$+ \sum_{m=1}^{M} \left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib} \frac{1}{\Omega} a_{SF_0} \cos \left(\Omega (t_{m-1} - t_0) + \Omega t_0 - \varphi_{a_{SF}} \right) \right] T_m \qquad (10.1.3.2.3\text{-}1)$$

$$\underline{R}_{SF_M} = - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \left[\sin \left(\Omega (t_M - t_0) + \Omega t_0 - \varphi_{a_{SF}} \right) - \sin \left(\Omega t_0 - \varphi_{a_{SF}} \right) \right]$$
$$+ \sum_{m=1}^{M} \left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib} \frac{1}{\Omega} a_{SF_0} \cos \left(\Omega (t_{m-1} - t_0) + \Omega t_0 - \varphi_{a_{SF}} \right) \right] T_m$$

$$\underline{v}_{SF_{m-1}} = \int_{t_0}^{t_{m-1}} \underline{a}_{SF}(t)\, dt$$

Let us now introduce a new folded frequency parameter as the difference between the actual vibration frequency Ω and an integer multiple of a base frequency determined by the l computation cycle rate. The l cycle update frequency is $1 / T_l$ cycles per second or $2 \pi / T_l$ radians per second. We define the folded frequency as the difference between Ω and its closest integer multiple of $2 \pi / T_l$ or:

$$k = \left(\frac{\Omega T_l}{2 \pi} \right)_{Intgr} \qquad \Omega' \equiv \Omega - \frac{2 \pi k}{T_l} \qquad (10.1.3.2.3\text{-}2)$$

where

k = Nearest integer value of the ratio of Ω to $2 \pi / T_l$.

$(\;)_{Intgr}$ = $(\;)$ rounded to the nearest integer value (e.g., $(0.3)_{Intgr} = 0$, $(0.5)_{Intgr} = 1$, $(0.7)_{Intgr} = 1$, $(1.3)_{Intgr} = 1$, $(1.5)_{Intgr} = 2$, $(1.7)_{Intgr} = 2$, etc.).

Ω' = Folded frequency.

Based on the definition for $(\;)_{Intgr}$ above, we see that:

$$-0.5 \leq \left[(\;) - (\;)_{Intgr} \right] < 0.5 \qquad (10.1.3.2.3\text{-}3)$$

Dividing the Ω' expression in (10.1.3.2.3-2) by $2\,\pi / T_l$ and substituting for k finds:

$$\frac{\Omega'\,T_l}{2\,\pi} = \frac{\Omega\,T_l}{2\,\pi} - \left(\frac{\Omega\,T_l}{2\,\pi}\right)_{Intgr} \tag{10.1.3.2.3-4}$$

Setting $\left[\,(\,) - (\,)_{Intgr}\right]$ in (10.1.3.2.3-3) to $\left[\frac{\Omega\,T_l}{2\,\pi} - \left(\frac{\Omega\,T_l}{2\,\pi}\right)_{Intgr}\right]$, we see from (10.1.3.2.3-3) and (10.1.3.2.3-4) that:

$$-\,0.5 \le \frac{\Omega'\,T_l}{2\,\pi} < 0.5 \tag{10.1.3.2.3-5}$$

or after multiplying by $2\,\pi$:

$$-\,\pi \le \Omega'\,T_l < \pi \tag{10.1.3.2.3-6}$$

Equation (10.1.3.2.3-6) identifies the limits for the Ω' folded frequency parameter defined by Equations (10.1.3.2.3-2).

Let us apply Ω' to Equations (10.1.3.2.3-1). First we write the (10.1.3.2.3-2) equivalent:

$$\Omega = \frac{2\,\pi\,k}{T_l} + \Omega' \tag{10.1.3.2.3-7}$$

Then from (10.1.3.2.2-38) and (10.1.3.2.2-18):

$$\frac{1}{T_l}(t_M - t_0) = \frac{T_m}{T_l}M = M\,r \qquad \frac{1}{T_l}(t_{m-1} - t_0) = \frac{T_m}{T_l}(m-1) = (m-1)\,r \tag{10.1.3.2.3-8}$$

Equation (10.1.3.2.3-7) multiplied by T_l and the result by (10.1.3.2.3-8) shows that:

$$\begin{aligned}
&\Omega\,T_l = 2\,\pi\,k + \Omega'\,T_l \\
&\Omega(t_M - t_0) = \frac{2\,\pi\,k}{T_l}(t_M - t_0) + \Omega'(t_M - t_0) = 2\,\pi\,k\,M\,r + \Omega'(t_M - t_0) \\
&\Omega(t_{m-1} - t_0) = \frac{2\,\pi\,k}{T_l}(t_{m-1} - t_0) + \Omega'(t_{m-1} - t_0) = 2\,\pi\,k\,(m-1)\,r + \Omega'(t_{m-1} - t_0)
\end{aligned} \tag{10.1.3.2.3-9}$$

Using (10.1.3.2.3-9) while noting that k, m and r are integers, we see that the sinusoidal terms in (10.1.3.2.3-1) are equivalently:

$$
\begin{aligned}
&\sin \Omega\, T_l \;=\; \sin \Omega'\, T_l \\
&\cos \Omega\, T_l \;=\; \cos \Omega'\, T_l \\
&\sin\left(\Omega(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \;=\; \sin\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \\
&\cos\left(\Omega(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \;=\; \cos\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \\
&\cos\left(\Omega(t_{m-1} - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \;=\; \cos\left(\Omega'(t_{m-1} - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right)
\end{aligned}
\qquad (10.1.3.2.3\text{-}10)
$$

Substituting (10.1.3.2.3-10) in (10.1.3.2.3-1) then obtains the equivalent result:

$$
\begin{aligned}
\underline{R}_{SF/Algo_M} \;=\; &- \underline{u}_{Vib}\, \frac{1}{\Omega^2}\, a_{SF_0} \left\{ \left(\frac{\Omega\, T_l \sin \Omega'\, T_l}{2\,(1 - \cos \Omega'\, T_l)} \right.\right. \\
&\left. + \frac{1}{12}\, \Omega\, T_l \sin \Omega'\, T_l \right) \left[\sin\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \right] \\
&\left. - \frac{1}{12}\, \Omega\, T_l \left[\cos\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \cos\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \right] (1 - \cos\, \Omega'\, T_l) \right\} \\
&+ \sum_{m=1}^{M} \left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib}\, \frac{1}{\Omega}\, a_{SF_0} \cos\left(\Omega'(t_{m-1} - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \right] T_m
\end{aligned}
\qquad (10.1.3.2.3\text{-}11)
$$

$$
\begin{aligned}
\underline{R}_{SF_M} \;=\; &- \underline{u}_{Vib}\, \frac{1}{\Omega^2}\, a_{SF_0} \left[\sin\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \right] \\
&+ \sum_{m=1}^{M} \left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib}\, \frac{1}{\Omega}\, a_{SF_0} \cos\left(\Omega'(t_{m-1} - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \right] T_m
\end{aligned}
\qquad (10.1.3.2.3\text{-}12)
$$

Equations (10.1.3.2.3-10) - (10.1.3.2.3-12) clearly show that the computer input data sampling process (at the l cycle rate) has the general effect on both the exact and INS algorithms of producing position solutions (at the computer sample times) whose sinusoidal components appear to be at the folded frequency Ω', rather than at the true position vibration frequency Ω. The terminology "folding" refers to the apparent mapping ("folding" back) of the frequency Ω sinusoidal components into the Ω' frequency band from minus to plus $\frac{\pi}{T_l}$ (see Equation (10.1.3.2.3-6)). We also note that the same is true for the attitude and velocity updating algorithms (both the exact and INS versions) as substitution of (10.1.3.2.3-10) for X and Y axis vibrations in the appropriate attitude/velocity solutions would reveal (i.e., for exact algorithm attitude in Equations (10.1.1.2.1-15), for INS algorithm attitude in (10.1.1.2.2-1) with (10.1.1.2.2-4), (10.1.1.2.1-1), (10.1.1.2.1-8) and (10.1.1.2.2-24), for exact algorithm velocity in Equation (10.1.2.2.1-12), and for INS algorithm velocity in Equations (10.1.2.2.2-1) with (10.1.2.2.2-5), (10.1.2.2.1-1), (10.1.2.2.1-2), (10.1.2.2.1-5) and (10.1.2.2.2-18)). In the case of the exact position algorithm, the solution so produced (i.e., $\underline{R}_{SF_M}$ in (10.1.3.2.3-12)) is the true solution at m cycle time M. However, for the INS position algorithm, the position solution (i.e., $\underline{R}_{SF/Algo_M}$ in (10.1.3.2.3-11)) may be distorted from $\underline{R}_{SF_M}$, particularly for the situation when the folded frequency Ω' approaches zero (i.e., when Ω is close to an integer multiple of $2\,\pi / T_l$ - See Equation (10.1.3.2.3-2), and $(1 - \cos\,\Omega' T_l)$ approaches zero - See Equation (10.1.3.2.3-11)). Let's look at the $\dfrac{\Omega\,T_l \sin\,\Omega' T_l}{2\,(1 - \cos\,\Omega' T_l)}$ term of concern in Equation (10.1.3.2.3-11) when Ω' approaches zero for two situations, $k = 0$ and $k > 0$.

From Equation (10.1.3.2.3-2), we see that the $k = 0$ situation corresponds to $\Omega' = \Omega$ and thus, with Equation (10.1.3.2.3-6), $\Omega\,T_l$ lying between zero and π (assuming we are only dealing with positive real Ω frequencies). Then the Ω' approaching zero condition can be stated as $\Omega\,T_l = \Omega' T_l = \varepsilon$ in which ε is a small number approaching zero. Under these conditions we can write:

$$\frac{\Omega\,T_l \sin\,\Omega' T_l}{2\,(1 - \cos\,\Omega' T_l)} = \frac{\varepsilon \sin\,\varepsilon}{2\,(1 - \cos\,\varepsilon)} = \frac{\varepsilon\,(\varepsilon - \dfrac{\varepsilon^3}{3\,!} + \cdots)}{2\,(1 - 1 + \dfrac{\varepsilon^2}{2\,!} - \cdots)} \approx \frac{\varepsilon^2}{\varepsilon^2} = 1 \qquad (10.1.3.2.3\text{-}13)$$

Similarly, we note that the coefficients for the $\frac{1}{12}\,\Omega\,T_l$ terms in (10.1.3.2.3-11) become $\Omega\,T_l \sin\,\Omega' T_l \approx \varepsilon^2$ and $\Omega\,T_l\,(1 - \cos\,\Omega' T_l) \approx \frac{1}{2}\,\varepsilon^3$ which, for small ε, are negligible compared to $\dfrac{\Omega\,T_l \sin\,\Omega' T_l}{2\,(1 - \cos\,\Omega' T_l)} \approx 1$. Substituting these findings in (10.1.3.2.3-11), we see that for the $k = 0$, $\Omega' = \Omega$ approaching zero situation, the $\underline{R}_{SF/Algo_M}$ solution becomes equal to the (10.1.3.2.3-12) $\underline{R}_{SF_M}$ true solution.

What about the case when $k > 0$ and Ω' approaches zero? Then we can also write $\Omega' T_l = \varepsilon$, but for this, from (10.1.3.2.3-7), $\Omega\,T_l = 2\,\pi\,k + \varepsilon$. Under these conditions we see that:

$$\frac{\Omega\, T_l \sin \Omega'\, T_l}{2\,(1 - \cos\, \Omega'\, T_l)} = \frac{(2\,\pi\, k + \varepsilon)\, \sin\, \varepsilon}{2\,(1 - \cos\, \varepsilon)}$$

$$= \frac{(2\,\pi\, k + \varepsilon)\,(\varepsilon - \frac{\varepsilon^3}{3\,!} + \cdots)}{2\,(1 - 1 + \frac{\varepsilon^2}{2\,!} - \cdots)} \approx \frac{2\,\pi\, k\, \varepsilon}{\varepsilon^2} = \frac{2\,\pi\, k}{\varepsilon} = \frac{2\,\pi\, k}{\Omega'\, T_l} \tag{10.1.3.2.3-14}$$

In the limit when $\Omega' = 0$, $\frac{\Omega\, T_l \sin \Omega'\, T_l}{2\,(1 - \cos\, \Omega'\, T_l)}$ and the Equation (10.1.3.2.3-11) $\underline{R}_{SF/Algo_M}$ amplitude go to infinity. Thus, the computed $\underline{R}_{SF/Algo_M}$ position has a potentially serious singularity condition that must be addressed when evaluating position algorithm performance. One obvious approach for avoiding the singularity all together is to operate the position algorithm at a high enough *l* cycle rate (i.e., small enough T_l) that Ω will be less than $\frac{2\,\pi}{T_l}$ over the range of possible Ω input vibration frequencies. From (10.1.3.2.3-2) we see that under these conditions k will never equal 2 or more, and Ω' will not approach zero for k = 1 provided that T_l is small enough that Ω is sufficiently smaller than $\frac{2\,\pi}{T_l}$ for the largest expected Ω. For safety, and to minimize $\frac{\Omega\, T_l \sin \Omega'\, T_l}{2\,(1 - \cos\, \Omega'\, T_l)}$ distortion from the ideal unity value (as in (10.1.3.2.3-12)), we might try setting T_l to one tenth of $\frac{2\,\pi}{\Omega}$ for the maximum Ω. However, in a real application, it may not be practical to operate the position algorithm at this high a rate. For example, it is not unusual to experience input vibration frequencies in actual usage that range up to 2 K Hz (or $2\,\pi\, 2000$ rad/sec). By the previous safety factor of 10 rule we would then require the position algorithm to operate at a T_l of $0.1 \times \frac{2\,\pi}{2\,\pi\, 2000}$ sec corresponding to an *l* cycle update frequency of 20 K Hz. This seems unreasonably high.

In the real world, we would operate the position algorithm at an *l* cycle update rate that may be slower than the maximum possible vibration frequency. In other words, Ω' may approach zero with k > 0 over ranges of the vibration frequencies; i.e., Equation (10.1.3.2.3-14) will apply, including the possibility for the $\Omega' = 0$ condition for which $\frac{\Omega\, T_l \sin \Omega'\, T_l}{2\,(1 - \cos\, \Omega'\, T_l)}$ is infinite.

What does this mean physically, and faced with this possibility, how can we use (10.1.3.2.3-14) to analyze the effect on $\underline{R}_{SF/Algo_M}$ performance?

Fortunately, the real world places practical limits on infinite analytical predictions. In this case, the obvious limit is the time it would take the $\underline{R}_{SF/Algo_M}$ position solution to reach the infinite value. From (10.1.3.2.3-11) we see that this would correspond to an infinite time because the sinusoid multiplied by the amplitude function containing $\frac{\Omega\, T_l \sin \Omega' T_l}{2\,(1 - \cos\, \Omega' T_l)}$, is at frequency Ω' which is zero for infinite $\frac{\Omega\, T_l \sin \Omega' T_l}{2\,(1 - \cos\, \Omega' T_l)}$. Thus, to find a finite $\underline{R}_{SF_M}$ solution near the $\Omega' = 0$ condition (for any k), $\underline{R}_{SF_M}$ must be evaluated over the finite navigation time.

For the remainder of this section we will explore an alternative version of the (10.1.3.2.3-11) $\underline{R}_{SF/Algo_M}$ solution that avoids the $\frac{\Omega\, T_l \sin \Omega' T_l}{2\,(1 - \cos\, \Omega' T_l)}$ singularity condition. The result will be an equation for $\underline{R}_{SF_M}$ that can be evaluated for any input and folded frequency, thereby providing a practical analysis tool for evaluating $\underline{R}_{SF/Algo_M}$ as a function of T_l. Application of the result enables the design engineer to assess the minimum T_l value required to meet a particular $\underline{R}_{SF/Algo_M}$ accuracy requirement over a specified navigation time period for a specified input vibration frequency and amplitude.

To develop the singularity free version of (10.1.3.2.3-11), we begin with the expansion:

$$\begin{aligned}
&\sin\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \\
&= \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \cos\left(\Omega'(t_M - t_0)\right) + \cos\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \sin\left(\Omega'(t_M - t_0)\right) \\
&\qquad - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \\
&= \cos\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \sin\left(\Omega'(t_M - t_0)\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \left[1 - \cos\left(\Omega'(t_M - t_0)\right)\right]
\end{aligned} \qquad (10.1.3.2.3\text{-}15)$$

Next we employ our functional friends from Equation (7.1.1.1-3) of Chapter 7 (and Equations (19.1.4-1) of Chapter 19):

$$\begin{aligned}
f_1(\vartheta) &\equiv \frac{\sin\, \vartheta}{\vartheta} = 1 - \frac{\vartheta^2}{3\,!} + \frac{\vartheta^4}{5\,!} - \cdots \\
f_2(\vartheta) &\equiv \frac{(1 - \cos\, \vartheta)}{\vartheta^2} = \frac{1}{2\,!} - \frac{\vartheta^2}{4\,!} + \frac{\vartheta^4}{6\,!} - \cdots
\end{aligned} \qquad (10.1.3.2.3\text{-}16)$$

where

$$f_1(\vartheta),\ f_2(\vartheta) \ = \text{Functional operators on the general angle parameter } \vartheta.$$

Based on the (10.1.3.2.3-16) definitions, we then write for particular Ω' trigonometric functions in (10.1.3.2.3-11) and (10.1.3.2.3-15):

$$\begin{aligned} &\sin \Omega' T_l \ = \ \Omega' T_l\, f_1(\Omega' T_l) \\ &(1 - \cos\ \Omega' T_l) \ = \ (\Omega' T_l)^2\, f_2(\Omega' T_l) \end{aligned} \qquad (10.1.3.2.3\text{-}17)$$

$$\begin{aligned} &\sin \left(\Omega' (t_M - t_0)\right) \ = \ \Omega'(t_M - t_0)\ f_1\left(\Omega'(t_M - t_0)\right) \\ &\left[1 - \cos \left(\Omega'(t_M - t_0)\right)\right] \ = \ \left(\Omega'(t_M - t_0)\right)^2 f_2\left(\Omega'(t_M - t_0)\right) \end{aligned} \qquad (10.1.3.2.3\text{-}18)$$

We now substitute (10.1.3.2.3-17) in the Equation (10.1.3.2.3-11) multiplier of $\sin\left(\Omega' (t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right)$; then substitute (10.1.3.2.3-18) in (10.1.3.2.3-15) with the result applied for $\sin\left(\Omega' (t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right)$ in (10.1.3.2.3-11). After factorization and rearrangement (left as an exercise for the reader), we obtain a singularity free form of $\underline{R}_{SF/Algo_M}$:

$$\begin{aligned} \underline{R}_{SF/Algo_M} \ = \ & - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \Bigg\{ \Omega(t_M - t_0) \left(\frac{f_1(\Omega' T_l)}{2\, f_2(\Omega' T_l)} + \frac{1}{12} (\Omega' T_l)^2\, f_1(\Omega' T_l) \right) \Big[\cos\left(\Omega\, t_0 \right. \\ & \left. - \varphi_{a_{SF}}\right) f_1\left(\Omega'(t_M - t_0)\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \Omega'(t_M - t_0)\ f_2\left(\Omega'(t_M - t_0)\right) \Big] \\ & - \frac{1}{12} \Omega\, T_l \left[\cos\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \cos\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \right] (1 - \cos\ \Omega' T_l) \Bigg\} \\ & + \sum_{m\,=\,1}^{M} \left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib} \frac{1}{\Omega} a_{SF_0} \cos\left(\Omega'(t_{m-1} - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \right] T_m \end{aligned} \qquad (10.1.3.2.3\text{-}19)$$

Comparing (10.1.3.2.3-19) with (10.1.3.2.3-11), we see that the fundamental difference is the replacement of the troublesome $\dfrac{\Omega\, T_l \sin\, \Omega' T_l}{2\,(1 - \cos\ \Omega' T_l)}$ term with $\dfrac{f_1(\Omega' T_l)}{2\, f_2(\Omega' T_l)}$. From the definitions of f_1 and f_2 in (10.1.3.2.3-16), we see that both are finite for all values of their arguments. Furthermore, from (10.1.3.2.3-6) and (10.1.3.2.3-16), we see that $f_2(\Omega' T_l)$ is non-zero over the range of $\Omega' T_l$, equaling $\dfrac{2}{\pi^2}$ at the $\Omega' T_l = \pm\, \pi$ ends and $\dfrac{1}{2}$ at the $\Omega' T_l = 0$ center.

Thus, $\dfrac{f_1(\Omega' T_l)}{2 f_2(\Omega' T_l)}$ is finite over the range of all possible Ω''s and Ω's, enabling $\underline{R}_{SF/Algo_M}$ to be evaluated with (10.1.3.2.3-19) for any vibration frequency and l cycle update time interval.

10.1.3.2.4 INS Position Algorithm Error Response

When analyzing the performance of the INS position algorithm, what we are really interested in is the algorithm error compared to the ideal error free position algorithm solution:

$$\delta\underline{R}_{SF/Algo_M} \equiv \underline{R}_{SF/Algo_M} - \underline{R}_{SF_M} \tag{10.1.3.2.4-1}$$

where

$$\delta\underline{R}_{SF/Algo_M} = \text{Error in } \underline{R}_{SF/Algo_M}.$$

An analytical expression for $\delta\underline{R}_{SF/Algo_M}$ in (10.1.3.2.4-1) is easily derived by subtracting (10.1.3.2.3-12) from (10.1.3.2.3-19). First we convert (10.1.3.2.3-12) to the (10.1.3.2.3-19) format by substituting(10.1.3.2.3-18) in (10.1.3.2.3-15) with the result applied in (10.1.3.2.3-12) for $\sin\left(\Omega'(t_M - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right)$:

$$\begin{aligned}\underline{R}_{SF_M} = & - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \left\{ \Omega(t_M - t_0) \frac{\Omega'}{\Omega} \left[\cos\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) f_1\left(\Omega'(t_M - t_0)\right) \right.\right. \\ & \left.\left. - \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \Omega'(t_M - t_0)\, f_2\left(\Omega'(t_M - t_0)\right) \right] \right\} \\ & + \sum_{m=1}^{M} \left[\underline{v}_{SF_{m-1}} + \underline{u}_{Vib} \frac{1}{\Omega} a_{SF_0} \cos\left(\Omega'(t_{m-1} - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) \right] T_m\end{aligned} \tag{10.1.3.2.4-2}$$

Subtracting (10.1.3.2.4-2) from (10.1.3.2.3-19) then obtains $\delta\underline{R}_{SF/Algo_M}$. Including (10.1.3.2.3-2) combined for completeness and replacing t_M with the general time variable t, yields the singularity free form:

$$\begin{aligned}\delta\underline{R}_{SF/Algo}(t) = & - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF_0} \left\{ \Omega(t - t_0) \left(\frac{f_1(\Omega' T_l)}{2\, f_2(\Omega' T_l)} - \frac{\Omega'}{\Omega} \right.\right. \\ & \left. + \frac{1}{12} \left(\Omega' T_l\right)^2 f_1\left(\Omega' T_l\right) \right) \left[\cos\left(\Omega\, t_0 - \varphi_{a_{SF}}\right)\; f_1\left(\Omega'(t - t_0)\right) \right.\end{aligned} \tag{10.1.3.2.4-3}$$

(Continued)

$$- \sin\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \Omega'(t - t_0)\, f_2\left(\Omega' (t - t_0)\right)\Big]$$

$$- \frac{1}{12}\, \Omega\, T_l \left[\cos\left(\Omega'(t - t_0) + \Omega\, t_0 - \varphi_{a_{SF}}\right) - \cos\left(\Omega\, t_0 - \varphi_{a_{SF}}\right) \right] \left(1 - \cos\, \Omega'\, T_l\right) \Bigg\}$$

$$\Omega' = \Omega - \frac{2\,\pi}{T_l} \left(\frac{\Omega\, T_l}{2\,\pi} \right)_{Intgr} \qquad \text{(10.1.3.2.4-3) (Continued)}$$

in which f_1 and f_2 are defined by Equations (10.1.3.2.3-16).

10.1.4 <u>SENSOR ERROR EFFECTS</u>

Several inertial sensor error mechanisms are excited by vibration rectification. In the subsections to follow we will address two types; when the individual sensor generates vibration rectification error, and when the sensor dynamic response characteristic produces vibration rectification error in the high speed strapdown computation algorithms.

10.1.4.1 INDIVIDUAL SENSOR VIBRATION RECTIFICATION

Two common individual accelerometer vibration rectification error sources are g-squared error and (for a pendulous accelerometer) anisoinertia error (Reference 31) which, under sinusoidal vibration, we can define analytically as:

$$\begin{aligned} \delta a_{Accl_{G2}}(t) &= L_{G2}\, a_{Accl_{Inpt}}(t)\;\; a_{Accl_{Pend}}(t) \\ \delta a_{Accl_{Aniso}}(t) &= L_{Aniso}\; \omega_{IB_{Accl/Inpt}}(t)\;\; \omega_{IB_{Accl/Pend}}(t) \end{aligned} \qquad \text{(10.1.4.1-1)}$$

with

$$\begin{aligned} a_{Accl_{Inpt}}(t) &= a_{Accl\,0_{Inpt}} \sin\left(\Omega\, t - \varphi_{a_{Accl\,Inpt}}\right) \\ a_{Accl_{Pend}}(t) &= a_{Accl\,0_{Pend}} \sin\left(\Omega\, t - \varphi_{a_{Accl\,Pend}}\right) \\ \omega_{IB_{Accl/Inpt}}(t) &= \omega_{IB0_{Accl/Inpt}} \sin\left(\Omega\, t - \varphi_{\omega_{IB_{Accl/Inpt}}}\right) \\ \omega_{IB_{Accl/Pend}}(t) &= \omega_{IB0_{Accl/Pend}} \sin\left(\Omega\, t - \varphi_{\omega_{IB_{Accl/Pend}}}\right) \end{aligned} \qquad \text{(10.1.4.1-2)}$$

where

$\delta a_{Accl_{G2}}(t)$, $\delta a_{Accl_{Aniso}}(t)$ = Accelerometer g-squared and anisoinertia error.

$a_{Accl_{Inpt}}(t)$, $a_{Accl_{Pend}}(t)$ = Specific force acceleration along the accelerometer input and pendulum axes.

L_{G2}, L_{Aniso} = Accelerometer g-squared and anisoinertia error coefficients.

$a_{Accl\,0_{Inpt}}, a_{Accl\,0_{Pend}}$ = Amplitudes of the $a_{Accl_{Inpt}}(t)$, $a_{Accl_{Pend}}(t)$ sinusoidal acceleration vibrations.

$\varphi_{a_{Accl\,Inpt}}, \varphi_{a_{Accl\,Pend}}$ = Phase angles for the $a_{Accl_{Inpt}}(t)$, $a_{Accl_{Pend}}(t)$ sinusoidal acceleration vibrations.

$\omega_{IB_{Accl/Inpt}}(t)$, $\omega_{IB_{Accl/Pend}}(t)$ = Components of accelerometer angular rotation rate relative to inertial space around the accelerometer input and pendulum axes.

$\omega_{IB0_{Accl/Inpt}}, \omega_{IB0_{Accl/Pend}}$ = Amplitudes of the $\omega_{IB_{Accl/Inpt}}(t)$, $\omega_{IB_{Accl/Pend}}(t)$ sinusoidal angular rate vibrations.

$\varphi_{\omega_{IB_{Accl/Inpt}}}, \varphi_{\omega_{IB_{Accl/Pend}}}$ = Phase angles for the $\omega_{IB_{Accl/Inpt}}(t)$, $\omega_{IB_{Accl/Pend}}(t)$ sinusoidal angular rate vibrations.

Ω = Vibration frequency (assumed equal for vibrations along and around the accelerometer input and pendulum axes). We are only studying the case when the vibration frequencies are equal. For unequal vibration frequencies between axes, analyses similar to those in Sections 10.1.1.1 and 10.1.2.1 would show that no error rectification occurs, hence, is only of academic interest.

The Equation (10.1.4.1-1) error mechanisms have a common structure that can be analyzed from the response of a general function:

$$f(t) = h_x(t)\ h_y(t) \tag{10.1.4.1-3}$$

with

$$h_x(t) = h_{0_x} \sin\left(\Omega\, t - \varphi_{h_x}\right) \qquad h_y(t) = h_{0_y} \sin\left(\Omega\, t - \varphi_{h_y}\right) \tag{10.1.4.1-4}$$

where

$f(t)$ = Generalized vibration induced sensor error.

x, y = Generalized axes associated with a particular inertial sensor.

$h_x(t)$, $h_y(t)$ = Generalized sinusoidal vibrations at frequency Ω along (around) the inertial sensor axes x and y.

$h_{0_x}, h_{0_y}, \varphi_{h_x}, \varphi_{h_y}$ = Amplitudes and phase angles associated with $h_x(t)$, $h_y(t)$.

We are concerned when f(t) develops a systematic (constant) component as manifested in a non-zero mean value over an extended time period which we define analytically as:

$$\overline{f(t)} \equiv \frac{1}{T}\int_{t_0}^{t_0+T} f(t)\ dt \quad {}_{T\to\infty} \tag{10.1.4.1-5}$$

where

$\overline{f(t)}$ = Mean value of f(t).

t_0 = Starting time for mean value evaluation.

T = Averaging time for mean value evaluation.

Combining (10.1.4.1-3) - (10.1.4.1-4) and expanding finds for the integrand in (10.1.4.1-5):

$$\begin{aligned} f(t) &= h_x(t)\ h_y(t) = h_{0_x}\, h_{0_y} \sin\left(\Omega\, t - \varphi_{h_x}\right) \sin\left(\Omega\, t - \varphi_{h_y}\right) \\ &= \frac{1}{2}\, h_{0_x}\, h_{0_y} \left\{\cos\left(\varphi_{h_y} - \varphi_{h_x}\right) - \cos\left[2\,\Omega\, t - \left(\varphi_{h_x} + \varphi_{h_y}\right)\right]\right\} \end{aligned} \tag{10.1.4.1-6}$$

With (10.1.4.1-6), the (10.1.4.1-5) integral is:

$$\begin{aligned} \frac{1}{T}\int_{t_0}^{t_0+T} f(t)\, dt &= \frac{1}{2}\, h_{0_x}\, h_{0_y} \Bigg\{\cos\left(\varphi_{h_y} - \varphi_{h_x}\right) \\ &- \frac{1}{2\,\Omega\, T} \sin\left[2\,\Omega\left(t_0 + T\right) - \left(\varphi_{h_x} + \varphi_{h_y}\right)\right] + \frac{1}{2\,\Omega\, T} \sin\left[2\,\Omega\, t_0 - \left(\varphi_{h_x} + \varphi_{h_y}\right)\right]\Bigg\} \end{aligned} \tag{10.1.4.1-7}$$

Letting T go to infinity in the limit then obtains for $\overline{f(t)}$:

$$\overline{f(t)} = \frac{1}{2}\, h_{0_x}\, h_{0_y} \cos\left(\varphi_{h_y} - \varphi_{h_x}\right) \tag{10.1.4.1-8}$$

Using (10.1.4.1-8), we can compare (10.1.4.1-1) - (10.1.4.1-2) to the general form (10.1.4.1-3) - (10.1.4.1-4) to write for the mean accelerometer g-squared and anisoinertia errors:

$$\begin{aligned} \overline{\delta a_{Accl_{G2}}} &= L_{G2}\, \frac{1}{2}\, a_{Accl\,0_{Inpt}}\, a_{Accl\,0_{Pend}} \cos\left(\varphi_{a_{Accl\,Pend}} - \varphi_{a_{Accl\,Inpt}}\right) \\ \overline{\delta a_{Accl_{Aniso}}} &= L_{Aniso}\, \frac{1}{2}\, \omega_{IB0_{Accl/Inpt}}\, \omega_{IB0_{Accl/Pend}} \cos\left(\varphi_{\omega_{IB_{Accl/Pend}}} - \varphi_{\omega_{IB_{Accl/Inpt}}}\right) \end{aligned} \tag{10.1.4.1-9}$$

where

$\overline{\delta a_{Accl_{G2}}}$, $\overline{\delta a_{Accl_{Aniso}}}$ = Mean values for $\delta a_{Accl_{G2}}(t)$, $\delta a_{Accl_{Aniso}}(t)$ defined similarly to (10.1.4.1-5).

10.1.4.2 VIBRATION RECTIFICATION FROM SENSOR DYNAMIC RESPONSE IN STRAPDOWN COMPUTATION ALGORITHMS

The strapdown computation algorithms developed in Chapters 7 and 19 (Section 19.1) were based on operating from angular rate sensor and accelerometer signals that provided an accurate measurement of B frame angular-rate/specific-force. Chapter 8 addressed the process by which raw outputs from the inertial sensors can be compensated to remove their predictable error mechanisms. Implicit to the Chapter 8 discussion was the assumption that the sensor outputs being compensated were a direct reflection of the inputs and errors, without dynamic time shift. In this section we introduce the problem created by the dynamic response of inertial sensors to vibration (high frequency) inputs in which the sensor output may be distorted in both amplitude and phase from the input. The result is a potential distortion in the output of the strapdown computation algorithms using the sensor data. In Section 10.2 we will discuss the details of describing the dynamic response characteristics for linear systems in general, and then apply them in subsequent sections to the results of this (and other) sections.

For simplicity, this section will only address the error in the velocity integration algorithms produced by inertial sensor dynamic response. The analysis methods utilized can then be applied to the effect of sensor dynamic response in the strapdown attitude integration algorithms. Let us proceed to define the problem at hand as the computation of velocity under sinusoidal vibration from angular rate sensors and accelerometers having the Section 10.1.2 <u>inputs</u>, and assuming, as usual, equal vibration frequencies for the vibration components (with the understanding that if the frequencies were different, no net deleterious error effects would arise):

$$\begin{aligned} \underline{\omega}_{IB}(t) &= \underline{u}_x \, \theta_{0_x} \, \Omega \cos\left(\Omega\, t - \varphi_{\theta_x}\right) \\ \underline{a}_{SF}(t) &= \underline{u}_y \, a_{SF0_y} \sin\left(\Omega\, t - \varphi_{a_{SF_y}}\right) \end{aligned} \qquad (10.1.4.2\text{-}1)$$

where

Ω = Vibration frequency.

In response to the (10.1.4.2-1) vibration input, the angular rate sensors and accelerometers provide the following outputs at the same frequency, but at a different amplitude and phase:

$$\begin{aligned} \underline{\omega}_{ARS}(t) &= \underline{u}_x \, \omega_{ARS0_x} \cos\left(\Omega\, t - \varphi_{\omega_{ARS_x}}\right) \\ \underline{a}_{Accl}(t) &= \underline{u}_y \, a_{Accl0_y} \sin\left(\Omega\, t - \varphi_{a_{Accl_y}}\right) \end{aligned} \qquad (10.1.4.2\text{-}2)$$

where

$\underline{\omega}_{ARS}(t)$, $\underline{a}_{Accl}(t)$ = Angular rate sensor and accelerometer outputs under the (10.1.4.2-1) vibration exposure.

$\omega_{ARS0_x}, a_{Accl0_y}$ = Amplitudes of the $\underline{\omega}_{ARS}(t)$, $\underline{a}_{Accl}(t)$ sinusoidal outputs.

$\varphi_{\omega_{ARS_x}}, \varphi_{a_{Accl_y}}$ = Phase angles for the $\underline{\omega}_{ARS}(t)$, $\underline{a}_{Accl}(t)$ sinusoidal outputs.

The (10.1.4.2-2) angular rate sensor and accelerometer output amplitudes and phase angles differ from those of the (10.1.4.2-1) input because of the sensor non-ideal dynamic response characteristics.

Section 10.1.2.1 (Equation (10.1.2.1-13)) showed that under the (10.1.4.2-1) vibration, a true constant velocity rate was generated along the B Frame Z axis identified as the sculling rate:

$$\dot{\underline{v}}_{SF_{Scul}} = \underline{u}_Z \, \frac{1}{2} \, \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \tag{10.1.4.2-3}$$

Therefore, the strapdown INS computation software under the (10.1.4.2-2) vibration input will generate a sculling rate equal to:

$$\dot{\underline{v}}_{SF/Scul_{SnsDyn}} = \underline{u}_Z \frac{1}{2} \, \omega_{ARS0_x} \frac{1}{\Omega} \, a_{Accl0_y} \cos\left(\varphi_{a_{Accl_y}} - \varphi_{\omega_{ARS_x}}\right) \tag{10.1.4.2-4}$$

where

$\dot{\underline{v}}_{SF/Scul_{SnsDyn}}$ = Sculling rate generated from INS software with input from inertial sensors having non-ideal dynamic response characteristics.

We can then define the velocity error caused by inertial sensor dynamic response as the difference between the true and INS computer generated velocity solutions:

$$\delta\dot{\underline{v}}_{SF/Scul_{SnsDyn}} \equiv \dot{\underline{v}}_{SF/Scul_{SnsDyn}} - \dot{\underline{v}}_{SF_{Scul}} \tag{10.1.4.2-5}$$

where

$\delta\dot{\underline{v}}_{SF/Scul_{SnsDyn}}$ = Error in $\dot{\underline{v}}_{SF/Scul_{SnsDyn}}$ due to non-ideal inertial sensor dynamic response characteristics.

Substituting (10.1.4.2-3) and (10.1.4.2-4) into (10.1.4.2-5) then finds for $\delta\dot{\underline{v}}_{SF/Scul_{SnsDyn}}$:

$$\delta\dot{\underline{v}}_{SF/Scul_{SnsDyn}} = \underline{u}_Z \, \frac{1}{2} \left[\omega_{ARS0_x} \frac{1}{\Omega} \, a_{Accl0_y} \cos\left(\varphi_{a_{Accl_y}} - \varphi_{\omega_{ARS_x}}\right) - \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \right] \tag{10.1.4.2-6}$$

or

$$\delta \dot{\underline{v}}_{SF/Scul_{SnsDyn}} = \underline{u}_Z \frac{1}{2\,\Omega} \Big[\omega_{ARS0_x}\, a_{Accl0_y} \cos\left(\varphi_{a_{Accl_y}} - \varphi_{\omega_{ARS_x}}\right) - \omega_{0_x}\, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right) \Big] \tag{10.1.4.2-7}$$

in which

$$\omega_{0_x} \equiv \theta_{0_x}\, \Omega \tag{10.1.4.2-8}$$

In general, the phase/amplitude response of an inertial sensor (with non-ideal dynamic response characteristics) to sinusoidal inputs tracks the input phase/amplitude for low frequencies (i.e., in Equation (10.1.4.2-4), $\varphi_{a_{Accl_y}} = \varphi_{a_{SF_y}}$, $\varphi_{\omega_{ARS_x}} = \varphi_{\theta_x}$, $\omega_{ARS0_x} = \omega_{0_x}$ and $a_{Accl0_y} = a_{SF0_y}$), and then loses tracking accuracy as the input frequency increases. We find, however, that the phase response with increasing frequency tends to vary from the input phase well before the amplitude varies from the input amplitude. An inertial sensor for strapdown inertial navigation applications should maintain good phase angle tracking for high input frequencies to minimize the sculling error demonstrated in Equation (10.1.4.2-7). (Note that the Equation (10.1.4.2-7) sculling error is zero when the inertial sensor output phase/amplitude equals the input phase/amplitude). However, it is also important to note from Equation (10.1.4.2-7), that it is the relative dynamic phase response between the accelerometers and angular rate sensors that affects sculling computation accuracy; not the individual phase angle response for each sensor. Based on this observation, one solution that can be considered for a dynamic mismatch in the accelerometer/angular-rate-sensor phase angle characteristic is to artificially add an output shaping filter to the wider bandwidth sensor to make its filtered output track the dynamic phase characteristic of the narrower bandwidth sensor. A similar discussion applies to the coning error generated for dynamically mismatched X/Y, Y/Z, and Z/X axis angular rate sensors, however, since all the angular rate sensors in a strapdown INS are generally of a common design, all will nominally have identical dynamic characteristics, and phase angle mismatch will only occur due to manufacturing variations between the axis components.

In subsequent sections, Equation (10.1.4.2-6) will be further analyzed based on analytical representations for the ω_{ARS0_x}, a_{Accl0_y}, $\varphi_{\omega_{ARS_x}}$, $\varphi_{a_{Accl_y}}$ dynamic response terms.

10.1.5 SUMMARY OF RESPONSE TO SINUSOIDAL SENSOR VIBRATION INPUT

Following is a summary of the principal sinusoidal response results derived in the previous subsections as provided by Equations (10.1.1.1-13), (10.1.1.2.1-14), (10.1.1.2.2-27), (10.1.1.2.2-28), (10.1.2.1-13), (10.1.2.2.1-11), (10.1.2.2.2-21), (10.1.2.2.2-22), (10.1.3.2.4-3), (10.1.4.1-8), (10.1.4.1-9), and (10.1.4.2-6):

$$\dot{\underline{\Phi}}_{Con} = \underline{u}_Z \frac{1}{2} \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)$$

$$\dot{\underline{\beta}}_m = \underline{u}_Z \frac{1}{2} \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)$$

$$\dot{\underline{\beta}}_{Algo_m} = \underline{u}_Z \frac{1}{2} \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos \Omega \, T_l\right)\right]\left(\frac{\sin \Omega \, T_l}{\Omega \, T_l}\right) - \frac{\sin \Omega \, T_m}{\Omega \, T_m}\right\}$$

$$\delta\dot{\underline{\Phi}}_{Algo_m} = \delta\dot{\underline{\beta}}_{Algo_m} =$$
$$\underline{u}_Z \frac{1}{2} \Omega \, \theta_{0_x} \, \theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos \Omega \, T_l\right)\right]\left(\frac{\sin \Omega \, T_l}{\Omega \, T_l}\right) - 1\right\}$$

$$\dot{\underline{v}}_{SF_{Scul}} = \underline{u}_Z \frac{1}{2} \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)$$

$$\Delta\dot{\underline{v}}_{Scul_m} = \underline{u}_Z \frac{1}{2} \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right) \tag{10.1.5-1}$$

$$\Delta\dot{\underline{v}}_{Scul/Algo_m} = \underline{u}_Z \frac{1}{2} \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left\{\left[1 + \right.\right.$$
$$\left.\left. + \frac{1}{3}\left(1 - \cos \Omega \, T_l\right)\right]\left(\frac{\sin \Omega \, T_l}{\Omega \, T_l}\right) - \frac{\sin \Omega \, T_m}{\Omega \, T_m}\right\}$$

$$\delta\dot{\underline{v}}_{SF/Algo_m} = \delta\Delta\dot{\underline{v}}_{Scul/Algo_m} =$$
$$\underline{u}_Z \frac{1}{2} \theta_{0_x} \, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos \Omega \, T_l\right)\right]\left(\frac{\sin \Omega \, T_l}{\Omega \, T_l}\right) - 1\right\}$$

$$\delta\underline{R}_{SF/Algo}(t) = - \underline{u}_{Vib} \frac{1}{\Omega^2} a_{SF0}\left\{\Omega(t - t_0)\left(\frac{f_1(\Omega' T_l)}{2 \, f_2(\Omega' T_l)} - \frac{\Omega'}{\Omega}\right.\right.$$
$$\left. + \frac{1}{12}\left(\Omega' T_l\right)^2 f_1\left(\Omega' T_l\right)\right)\left[\cos\left(\Omega \, t_0 - \varphi_{a_{SF}}\right) f_1\left(\Omega'(t - t_0)\right)\right.$$
$$\left. - \sin\left(\Omega \, t_0 - \varphi_{a_{SF}}\right) \Omega'(t - t_0) \, f_2\left(\Omega'(t - t_0)\right)\right]$$
$$\left. - \frac{1}{12}\Omega \, T_l\left[\cos\left(\Omega'(t - t_0) + \Omega \, t_0 - \varphi_{a_{SF}}\right) - \cos\left(\Omega \, t_0 - \varphi_{a_{SF}}\right)\right]\left(1 - \cos \Omega' T_l\right)\right\}$$

(Continued)

$$\Omega' = \Omega - \frac{2\,\pi}{T_l}\left(\frac{\Omega\,T_l}{2\,\pi}\right)_{Intgr} \qquad f_1 \text{ and } f_2 \text{ are defined by Equations (10.1.3.2.3-16).}$$

$$\overline{f(t)} = \frac{1}{2}\,h0_x\,h0_y\,\cos\left(\varphi_{h_y} - \varphi_{h_x}\right) \tag{10.1.5-1 (Continued)}$$

$$\overline{\delta a_{Accl_{G2}}} = L_{G2}\,\frac{1}{2}\,aAccl0_{Inpt}\,aAccl0_{Pend}\,\cos\left(\varphi_{aAccl_{Pend}} - \varphi_{aAccl_{Inpt}}\right)$$

$$\overline{\delta a_{Accl_{Aniso}}} = L_{Aniso}\,\frac{1}{2}\,\omega IB0_{Accl/Inpt}\,\omega IB0_{Accl/Pend}\,\cos\left(\varphi_{\omega IB_{Accl/Pend}} - \varphi_{\omega IB_{Accl/Inpt}}\right)$$

$$\delta\dot{\underline{v}}_{SF/Scul_{SnsDyn}} = \underline{u}_Z\,\frac{1}{2}\left[\omega ARS0_x\,\frac{1}{\Omega}\,aAccl0_y\,\cos\left(\varphi_{aAccl_y} - \varphi_{\omega ARS_x}\right) - \theta 0_x\,aSF0_y\,\cos\left(\varphi_{aSF_y} - \varphi_{\theta_x}\right)\right]$$

10.2 REVIEW OF LINEAR DYNAMIC FREQUENCY RESPONSE ANALYTICS

In Section 10.1 we analyzed the effect of sinusoidal angular rate and linear acceleration vibrations at discrete frequencies on strapdown INS computational performance. For the cases investigated, we have presumed knowledge of the amplitude and phase of the inertial sensor angular-rate/linear-acceleration output measurements; either equal to the sensor inputs (i.e., for idealized error free instruments) or at the input frequency with a specified output amplitude and phase angle. In general, the sensor vibration inputs are generated by vehicle vibrations translating through mechanical structural resonances connecting the sensors to the vibration sources. Furthermore, the vibration sources generally consist of a spectrum of frequency components at differing energy levels. In this section we will review linear dynamic system theory to develop the analytic tools necessary to translate the vibration source inputs into their ultimate effect on inertial sensor input and output. We will then apply the theory in subsequent sections to extend the Section 10.1 results to include system dynamic response effects to discrete sinusoidal and random input source vibrations. It is assumed in the development to follow that the reader has a working understanding of Laplace transforms.

10.2.1 <u>LINEAR SYSTEM RESPONSE TO SINUSOIDAL INPUTS</u>

Consider a general linear process that satisfies the differential equation:

$$y(t) + c_{y_1}\,\dot{y}(t) + c_{y_2}\,\ddot{y}(t) + \cdots = c_{x_0}\,x(t) + c_{x_1}\,\dot{x}(t) + c_{x_2}\,\ddot{x}(t) + \cdots \tag{10.2.1-1}$$

where

c_{x_i}, c_{y_i} = Constants.

$x(t), y(t)$ = Process input and output.

The "particular solution" to (10.2.1-1) for y(t) is its response to x(t), ignoring initial conditions on y(t) and its derivatives. The Laplace transform of the (10.2.1-1) particular solution is obtained as the Laplace transform of (10.2.1-1) with the y, $\dot{y}$, $\ddot{y}$: etc. initial condition terms ignored (Reference 10 - Appendix II, Laplace Transform Pair Number 6):

$$Y(S) = H(S)\, X(S) \tag{10.2.1-2}$$

with

$$H(S) = \frac{c_{x_0} + c_{x_1} S + c_{x_2} S^2 + \cdots}{1 + c_{y_1} S + c_{y_2} S^2 + \cdots} \tag{10.2.1-3}$$

where

S = Laplace transform parameter.

$X(S)$ = Laplace transform of x(t).

$Y(S)$ = Laplace transform of the y(t) response to x(t).

$H(S)$ = Transfer function defining the Y(S) response to the X(S) input. Since the Laplace transform of a unit impulse is one, we see from (10.2.1-2) that the Y(S) response to a unit impulse is H(S). Thus, the transfer function H(S) is the Laplace transform of the y(t) impulse response.

Let us consider the Y(S) response to a general x(t) sinusoid of the form:

$$x(t) = A \sin(\omega\, t + \psi) \tag{10.2.1-4}$$

where

ω, A, ψ = Frequency, amplitude, phase angle of the x(t) sinusoid, each considered constant.

The Laplace transform of (10.2.1-4) is (Reference 10 - Appendix III, Laplace Transform Pair Number 104):

$$X(S) = \frac{A\left(\omega \cos\psi + S \sin\psi\right)}{S^2 + \omega^2} \tag{10.2.1-5}$$

and the Y(S) response from (10.2.1-2) will be:

$$Y(S) \;=\; H(S)\;\frac{A\left(\omega \cos\,\psi + S\,\sin\,\psi\right)}{S^2 + \omega^2} \tag{10.2.1-6}$$

Equation (10.2.1-6) can be expanded as a linear function of its characteristic roots:

$$Y(S) \;=\; \frac{E_1}{S - r_1} + \frac{E_2}{S - r_2} + \frac{E_3}{S - r_3} + \cdots + \frac{D_1}{S + j\,\omega} + \frac{D_2}{S - j\,\omega} \tag{10.2.1-7}$$

where

r_i = Characteristic root of the H(S) denominator.

E_i, D_i = Constants.

j = Imaginary number equal to the square root of minus one.

Equation (10.2.1-7) is based on H(S) having no equal denominator roots. If some of the r_i's are equal, (10.2.1-7) will be a little more complicated, however, the results to follow will not be affected. The E_i terms in (10.2.1-7) correspond to time functions (inverse Laplace transforms) of the form $E_i\, e^{r_i\, t}$ (Reference 10 - Appendix III, Laplace Transform Pair Number 9). For a stable process, the r_i's will be negative, hence, the $E_i\, e^{r_i\, t}$ terms will be transitory, decaying to zero. What will be left, then, are the two D_i terms. Let us analyze these terms in more detail.

The magnitude of each D_i can be determined by multiplying (10.2.1-7) by the D_i denominator and setting the denominator to zero. For the D_2 coefficient, setting the denominator to zero corresponds to setting S = jω, hence:

$$Y(S)\,\left(S - j\omega\right)\Big]_{S \to j\omega} \;=$$

$$\left(\frac{E_1}{S - r_1} + \frac{E_2}{S - r_2} + \frac{E_3}{S - r_3} + \cdots + \frac{D_1}{S + j\,\omega}\right)\left(S - j\omega\right)\Bigg]_{S \to j\omega} + D_2 \;=\; D_2 \tag{10.2.1-8}$$

or with (10.2.1-6) for Y(S):

$$D_2 \;=\; H(S)\,\frac{A\left(\omega \cos\,\psi + S\,\sin\,\psi\right)}{S^2 + \omega^2}\left(S - j\omega\right)\Bigg]_{S \to j\omega} \tag{10.2.1-9}$$

$$=\; H(S)\,\frac{A\left(\omega \cos\,\psi + S\,\sin\,\psi\right)}{\left(S + j\omega\right)\left(S - j\omega\right)}\left(S - j\omega\right)\Bigg]_{S \to j\omega} \;=\; H(j\omega)\,\frac{A\left(\omega \cos\,\psi + j\,\omega\,\sin\,\psi\right)}{2\,j\omega}$$

hence:

$$D_2 = \frac{A\left(\cos\psi + j\sin\psi\right)}{2\,j} H(j\omega) \tag{10.2.1-10}$$

Similarly, we find for D_1:

$$D_1 = -\frac{A\left(\cos\psi - j\sin\psi\right)}{2\,j} H(-j\omega) \tag{10.2.1-11}$$

Thus, D_1 is the complex conjugate of D_2.

Further analysis will be expedited if we introduce the well known mathematical identity (Euler's theorem):

$$e^{j\,\psi} = \cos\psi + j\sin\psi \tag{10.2.1-12}$$

Then (10.2.1-10) and (10.2.1-11) simplify to:

$$D_1 = -\frac{A}{2\,j} e^{-j\psi} H(-j\omega) \qquad D_2 = \frac{A}{2\,j} e^{j\psi} H(j\omega) \tag{10.2.1-13}$$

Let us now analyze the form of the transfer function $H(j\omega)$ in (10.2.1-13) for which we can write in general:

$$H(j\omega) = Re(\omega) + j\,Im(\omega) \tag{10.2.1-14}$$

where

$Re(\omega), Im(\omega)$ = Real numbers corresponding to the real and imaginary portions of $H(j\omega)$.

We could have also defined the equivalent form:

$$H(j\omega) = B(\omega)\cos\phi(\omega) + j\,B(\omega)\sin\phi(\omega) \tag{10.2.1-15}$$

which from (10.2.1-14) has the equivalencies:

$$B(\omega)\cos\phi(\omega) = Re(\omega) \qquad B(\omega)\sin\phi(\omega) = Im(\omega) \tag{10.2.1-16}$$

where

$B(\omega), \phi(\omega)$ = Amplitude ratio and phase angle associated with $H(j\,\omega)$.

Taking the square root of the sum of the squares of the (10.2.1-16) and the arc tangent of the ratio then provides the relationship between $B(\omega)$, $\phi(\omega)$ and $Re(\omega)$, $Im(\omega)$:

$$B(\omega) = \sqrt{Re(\omega)^2 + Im(\omega)^2} \qquad \phi(\omega) = \tan^{-1}\left(\frac{Im(\omega)}{Re(\omega)}\right) \tag{10.2.1-17}$$

The Equation (10.2.1-15) form will prove more useful for our purposes because we can then apply our (10.2.1-12) formula for simplification:

$$H(j\omega) = B(\omega)\left(\cos\phi(\omega) + j\sin\phi(\omega)\right) = B(\omega)\, e^{j\,\phi(\omega)} \tag{10.2.1-18}$$

Similarly we find for the $H(-j\omega)$ term in (10.2.1-13):

$$H(-j\omega) = B(\omega)\, e^{-j\,\phi(\omega)} \tag{10.2.1-19}$$

Now substitute (10.2.1-18) - (10.2.1-19) into (10.2.1-13) to obtain for the D_i coefficients:

$$D_1 = -\frac{A}{2j} e^{-j\psi} B(\omega)\, e^{-j\phi(\omega)} = -\frac{A}{2j} B(\omega)\, e^{-j\left[\phi(\omega)+\psi\right]} \tag{10.2.1-20}$$

$$D_2 = \frac{A}{2j} e^{j\psi} B(\omega)\, e^{j\phi(\omega)} = \frac{A}{2j} B(\omega)\, e^{j\left[\phi(\omega)+\psi\right]} \tag{10.2.1-21}$$

Equations (10.2.1-20) and (10.2.1-21) are in a convenient form for substitution back into (10.2.1-7). First, however, let us state two other mathematical identities stemming from Equation (10.2.1-12):

$$\sin\psi = \frac{1}{2j}\left(e^{j\psi} - e^{-j\psi}\right) \qquad \cos\psi = \frac{1}{2}\left(e^{j\psi} + e^{-j\psi}\right) \tag{10.2.1-22}$$

Let us now proceed back to equation (10.2.1-7) in which the D_i terms can be combined into the following form:

$$\frac{D_1}{S + j\,\omega} + \frac{D_2}{S - j\,\omega} = \frac{D_1\left(S - j\,\omega\right) + D_2\left(S + j\,\omega\right)}{S^2 + \omega^2} = \frac{\left(D_2 + D_1\right) S + j\left(D_2 - D_1\right)\omega}{S^2 + \omega^2} \tag{10.2.1-23}$$

The $D_2 + D_1$ and $j\left(D_2 - D_1\right)$ terms in (10.2.1-23) are then easily evaluated from (10.2.1-20) - (10.2.1-22) to be:

$$D_2 + D_1 = A\,B(\omega)\,\frac{1}{2\,j}\left(e^{\,j\left[\phi(\omega)+\psi\right]} - e^{-\,j\left[\phi(\omega)+\psi\right]}\right) = A\,B(\omega)\,\sin\left(\phi(\omega)+\psi\right)$$
$$j\left(D_2 - D_1\right) = A\,B(\omega)\,\frac{1}{2}\left(e^{\,j\left[\phi(\omega)+\psi\right]} + e^{-\,j\left[\phi(\omega)+\psi\right]}\right) = A\,B(\omega)\,\cos\left(\phi(\omega)+\psi\right) \qquad (10.2.1\text{-}24)$$

Finally, we substitute (10.2.1-23) with (10.2.1-24) into Equation (10.2.1-7) to obtain for Y(S):

$$Y(S) = \frac{E_1}{S - r_1} + \frac{E_2}{S - r_2} + \frac{E_3}{S - r_3} + \cdots + \frac{B(\omega)\,A\left[\omega\cos\left(\phi(\omega)+\psi\right) + S\,\sin\left(\phi(\omega)+\psi\right)\right]}{S^2 + \omega^2} \qquad (10.2.1\text{-}25)$$

Equation (10.2.1-25) is in a form that enables a simple inverse Laplace transformation back to the time domain. Using Equations (10.2.1-4) and (10.2.1-5) as a guide, the result is:

$$y_x(t) = \text{Transitory terms} + B(\omega)\,A\,\sin\left(\omega\,t + \psi + \phi(\omega)\right)$$
for
$$x(t) = A\,\sin\left(\omega\,t + \psi\right) \qquad (10.2.1\text{-}26)$$

where

$y_x(t)$ = Portion of y(t) generated by x(t).

The transitory terms in (10.2.1-26) decay to zero for a stable process. Thus, Equation (10.2.1-26) shows that the "steady state" response of y(t) to a sinusoidal x(t) input at frequency ω is to generate a sinusoidal output at the same frequency ω, but with amplitude multiplied by $B(\omega)$ and phase increased by $\phi(\omega)$. The $B(\omega)$, $\phi(\omega)$ factors represent the dynamic response characteristics of the Equation (10.2.1-1) linear process as defined analytically by (10.2.1-14) and (10.2.1-17) for the H(S) transfer function.

Equation (10.2.1-17) for $B(\omega)$, $\phi(\omega)$ is generally not a convenient form for analytical evaluation. For the remainder of this section we will develop an expanded version of this equation that enables $B(\omega)$, $\phi(\omega)$ to be calculated from individual contributors to the H(S) transfer function.

Consider a linear process formed from a series of processes of the general form given by (10.2.1-1). For example, let's say $y_1(t)$ is generated from x(t) in a process of the general (10.2.1-1) form, then $y_2(t)$ is generated from $y_1(t)$ using the (10.2.1-1) format but with $y_1(t)$ input and $y_2(t)$ output (with different coefficients), $y_3(t)$ is formed from $y_2(t)$, etc., eventually leading to y(t). Each of these processes (i.e., x(t) to $y_1(t)$, $y_1(t)$ to $y_2(t)$, $y_2(t)$ to $y_3(t)$, $\cdots$ to y(t)) will have a transfer function and Laplace transform relating output to input of the form given in (10.2.1-2) and (10.2.1-3). Thus:

$$\begin{aligned} Y_1(S) &= H_1(S)\,X(S) \\ Y_2(S) &= H_2(S)\;Y_1(S) \\ Y_3(S) &= H_3(S)\;Y_2(S) \\ &\;\;\vdots \end{aligned} \tag{10.2.1-27}$$

where

$H_i(S)$ = Transfer function of intermediate linear process.

Then the composite of (10.2.1-27) is:

$$Y(S) = H_1(S)\;H_2(S)\;H_3(S)\cdots X(S) \tag{10.2.1-28}$$

hence, with (10.2.1-2):

$$H(S) = H_1(S)\;H_2(S)\;H_3(S)\cdots \tag{10.2.1-29}$$

Consider that each of the transfer functions in (10.2.1-29) has a numerator and denominator polynomial in S as in Equation (10.2.1-3):

$$H_i(S) = \frac{H_{Num_i}(S)}{H_{Den_i}(S)} \tag{10.2.1-30}$$

where

$H_{Num_i}(S)$, $H_{Den_i}(S)$ = Numerator and denominator polynomials of $H_i(S)$.

Substituting jω for S with (10.2.1-30) in (10.2.1-29) then obtains:

$$H(j\omega) = \frac{H_{Num\,1}(j\omega)\;H_{Num\,2}(j\omega)\;H_{Num\,3}(j\omega)\cdots}{H_{Den_1}(j\omega)\;H_{Den_2}(j\omega)\;H_{Den_3}(j\omega)\cdots} \tag{10.2.1-31}$$

Using (10.2.1-14), (10.2.1-17) and (10.2.1-18) as a general framework, the $H_{Num\,i}(S)$, $H_{Den_i}(S)$ terms in (10.2.1-31) can be converted to the equivalent forms:

$$
\begin{aligned}
H_{Num_i}(j\omega) &= Re_{Num_i}(\omega) + j\, Im_{Num_i}(\omega) = B_{Num_i}(\omega)\, e^{j\,\phi_{Num_i}(\omega)} \\
B_{Num_i}(\omega) &= \sqrt{Re^2_{Num_i}(\omega) + Im^2_{Num_i}(\omega)} \\
\phi_{Num_i}(\omega) &= \tan^{-1} \frac{Im_{Num_i}(\omega)}{Re_{Num_i}(\omega)} \\
H_{Den_i}(j\omega) &= Re_{Den_i}(\omega) + j\, Im_{Den_i}(\omega) = B_{Den_i}(\omega)\, e^{j\,\phi_{Den_i}(\omega)} \qquad (10.2.1\text{-}32) \\
B_{Den_i}(\omega) &= \sqrt{Re^2_{Den_i}(\omega) + Im^2_{Den_i}(\omega)} \\
\phi_{Den_i}(\omega) &= \tan^{-1} \frac{Im_{Den_i}(\omega)}{Re_{Den_i}(\omega)}
\end{aligned}
$$

where

$Re_{Num_i}(\omega)$, $Im_{Num_i}(\omega)$, $Re_{Den_i}(\omega)$, $Im_{Den_i}(\omega)$ = Real numbers corresponding to the real and imaginary portions of $H_{Num_i}(j\omega)$, $H_{Den_i}(j\omega)$.

$B_{Num_i}(\omega)$, $\phi_{Num_i}(\omega)$, $B_{Den_i}(\omega)$, $\phi_{Den_i}(\omega)$ = Amplitude ratios and phase angles associated with $H_{Num_i}(j\omega)$, $H_{Den_i}(j\omega)$.

Substituting (10.2.1-32) in (10.2.1-31) then yields for $H(j\omega)$:

$$
H(j\omega) = \frac{B_{Num_1}(\omega)\, B_{Num_2}(\omega)\, B_{Num_3}(\omega) \cdots e^{j\left(\phi_{Num_1}(\omega) + \phi_{Num_2}(\omega) + \phi_{Num_3}(\omega) \cdots\right)}}{B_{Den_1}(\omega)\, B_{Den_2}(\omega)\, B_{Den_3}(\omega) \cdots e^{j\left(\phi_{Den_1}(\omega) + \phi_{Den_2}(\omega) + \phi_{Den_3}(\omega) \cdots\right)}} \qquad (10.2.1\text{-}33)
$$

$$
= \frac{B_{Num_1}(\omega)\, B_{Num_2}(\omega)\, B_{Num_3}(\omega) \cdots}{B_{Den_1}(\omega)\, B_{Den_2}(\omega)\, B_{Den_3}(\omega) \cdots}\, e^{j\left(\phi_{Num_1}(\omega) + \phi_{Num_2}(\omega) + \cdots - \phi_{Den_1}(\omega) - \phi_{Den_2}(\omega) - \cdots\right)}
$$

Finally, we equate the (10.2.1-33) result to (10.2.1-18) to obtain formula for the $H(j\omega)$ amplitude and phase angle as a function of the individual numerator/denominator component amplitudes and phase angles:

$$
B(\omega) = \frac{B_{Num_1}(\omega)\, B_{Num_2}(\omega)\, B_{Num_3}(\omega) \cdots}{B_{Den_1}(\omega)\, B_{Den_2}(\omega)\, B_{Den_3}(\omega) \cdots} \qquad (10.2.1\text{-}34)
$$

$$
\phi(\omega) = \phi_{Num_1}(\omega) + \phi_{Num_2}(\omega) + \phi_{Num_3}(\omega) + \cdots - \phi_{Den_1}(\omega) - \phi_{Den_2}(\omega) - \phi_{Den_3}(\omega) - \cdots
$$

10.2.2 LINEAR SYSTEM RESPONSE TO RANDOM INPUTS

Section 10.2.1 dealt with the response of linear systems to sinusoidal inputs at a particular frequency. In this section we address the more general problem of describing linear system response to general time function inputs that are random from time function sample to sample. This will be accomplished by building on the results of the previous section using the Fourier series approach (Reference 10 - Section 2.1) for describing a general time function over a sample time interval:

$$p(t) = \sum_{i=0}^{\infty} p_i(t)$$

$$p_i(t) = a_i \cos(i\,\Delta\omega\,t) + b_i \sin(i\,\Delta\omega\,t)$$

$$\Delta\omega = \frac{2\,\pi}{T}$$

$$a_0 = \frac{1}{T}\int_{-0.5\,T}^{0.5\,T} p(t)\,dt \qquad b_0 = 0 \qquad (10.2.2\text{-}1)$$

$$a_i = \frac{2}{T}\int_{-0.5\,T}^{0.5\,T} p(t)\cos(i\,\Delta\omega\,t)\,dt \quad \text{for } i > 0$$

$$b_i = \frac{2}{T}\int_{-0.5\,T}^{0.5\,T} p(t)\sin(i\,\Delta\omega\,t)\,dt \quad \text{for } i > 0$$

where

i = Index for terms in the Fourier series.

$p(t)$ = General time function.

T = Width of the time interval over which the Fourier series fits the time function. The particular form of the Fourier series in (10.2.2-1) is based on the time interval extending from $t = -\frac{T}{2}$ to $t = \frac{T}{2}$.

$\Delta\omega$ = Base frequency for the Fourier series corresponding to the time interval T.

$p_i(t)$ = Component i of the general Fourier series consisting of a sinusoid at frequency $i\,\Delta\omega$.

a_i, b_i = Fourier series coefficients in $p_i(t)$.

For our purposes, the mean squared value of p(t) will be of primary importance which we define as a linear average of p(t) squared over the Fourier series time interval:

$$\overline{p(t)^2} = \frac{1}{T}\int_{-0.5\,T}^{0.5\,T} p(t)^2\,dt \tag{10.2.2-2}$$

where

$\overline{p(t)^2}$ = p(t) mean squared value.

If we substitute p(t) from (10.2.2-1) into (10.2.2-2) and analytically carry out the integration operations, we can determine an expression for $\overline{p(t)^2}$ as a function of the Fourier coefficients:

$$\overline{p(t)^2} = \frac{1}{T}\int_{-0.5\,T}^{0.5\,T} \sum_{i=0}^{\infty} p_i(t) \sum_{j=0}^{\infty} p_j(t)\,dt$$

$$= \frac{1}{T}\int_{-0.5\,T}^{0.5\,T} \left[\sum_{i=0}^{\infty}\Big(a_i \cos(i\,\Delta\omega\,t) + b_i \sin(i\,\Delta\omega\,t)\Big)\sum_{j=0}^{\infty}\Big(a_j \cos(j\,\Delta\omega\,t) + b_j \sin(j\,\Delta\omega\,t)\Big)\right] dt$$

$$= \frac{1}{T}\int_{-0.5\,T}^{0.5\,T} \Bigg[\sum_{i=0}^{\infty}\Big(a_i^2 \cos^2(i\,\Delta\omega\,t) + b_i^2 \sin^2(i\,\Delta\omega\,t)\Big)$$

$$+ \sum_{i=0}^{\infty}\sum_{\substack{j=0 \\ j\neq i}}^{\infty}\Big(a_i\,a_j \cos(i\,\Delta\omega\,t)\cos(j\,\Delta\omega\,t) + a_i\,b_j \cos(i\,\Delta\omega\,t)\sin(j\,\Delta\omega\,t) \tag{10.2.2-3}$$

$$+ b_i\,a_j \sin(i\,\Delta\omega\,t)\cos(j\,\Delta\omega\,t) + b_i\,b_j \sin(i\,\Delta\omega\,t)\sin(j\,\Delta\omega\,t)\Big)\Bigg] dt$$

$$= \frac{1}{T}\int_{-0.5\,T}^{0.5\,T} \sum_{i=0}^{\infty}\Big(a_i^2 \cos^2(i\,\Delta\omega\,t) + b_i^2 \sin^2(i\,\Delta\omega\,t)\Big)\,dt$$

$$= \sum_{i=0}^{\infty}\frac{1}{T}\int_{-0.5\,T}^{0.5\,T}\Big(a_i^2 \cos^2(i\,\Delta\omega\,t) + b_i^2 \sin^2(i\,\Delta\omega\,t)\Big)\,dt$$

where

j = Substitute (for i) Fourier series index.

or

$$\overline{p(t)^2} = \sum_{i=0}^{\infty} \frac{1}{2}\left(a_i^2 + b_i^2\right) \tag{10.2.2-4}$$

Let us now substitute for a_i, b_i from (10.2.2-1) to develop an equation for the $\frac{1}{2}\left(a_i^2 + b_i^2\right)$'s in (10.2.2-4) as a function of p(t) and the i $\Delta\omega$ frequency associated with each i:

$$\frac{1}{2}\left(a_0^2 + b_0^2\right) = \frac{1}{2\,T^2}\int_{-0.5\,T}^{0.5\,T} p(t)\,dt \int_{-0.5\,T}^{0.5\,T} p(u)\,du$$

$$= \frac{1}{2\,T^2}\int_{-0.5\,T}^{0.5\,T}\int_{-0.5\,T}^{0.5\,T} p(t)\,p(u)\,dt\,du$$

For i > 0:

$$\frac{1}{2}\left(a_i^2 + b_i^2\right) = \frac{2}{T^2}\left\{\int_{-0.5\,T}^{0.5\,T} p(t)\cos(i\,\Delta\omega\,t)\,dt \int_{-0.5\,T}^{0.5\,T} p(u)\cos(i\,\Delta\omega\,u)\,du \right. \tag{10.2.2-5}$$

$$\left. + \int_{-0.5\,T}^{0.5\,T} p(t)\sin(i\,\Delta\omega\,t)\,dt \int_{-0.5\,T}^{0.5\,T} p(u)\sin(i\,\Delta\omega\,u)\,du \right\}$$

$$= \frac{2}{T^2}\int_{-0.5\,T}^{0.5\,T}\int_{-0.5\,T}^{0.5\,T} p(t)\,p(u)\left(\cos(i\,\Delta\omega\,t)\cos(i\,\Delta\omega\,u) + \sin(i\,\Delta\omega\,t)\sin(i\,\Delta\omega\,u)\right)dt\,du$$

$$= \frac{2}{T^2}\int_{-0.5\,T}^{0.5\,T}\int_{-0.5\,T}^{0.5\,T} p(t)\,p(u)\cos\left(i\,\Delta\omega\,(u - t)\right)dt\,du$$

where

u = Substitute (for t) running time parameter.

At this point we introduce the concept of a random process as an ensemble of p(t)'s representing a potential group of p(t) histories running in parallel over time t. In fact, of course, only one of the p(t)'s (randomly selected) will actually exist in the real world. We also hypothesize that each p(t) ensemble member is random in some analytically defined manner relative to the other members of the ensemble. Analyses can then be performed over the ensemble at particular time points to statistically characterize p(t). For example, we will be dealing with the expected value of the p(t) mean squared value which from (10.2.2-4) can be written as:

$$\mathcal{E}\left(\overline{p(t)^2}\right) = \sum_{i=0}^{\infty} \mathcal{E}\left[\frac{1}{2}\left(a_i^2 + b_i^2\right)\right] \tag{10.2.2-6}$$

where

$\mathcal{E}(\ )$ = Expected value operator representing the average of ( ) over the ensemble members at some time point t.

Continued development of Equations (10.2.2-5) in the statistical ensemble world is expedited by introducing the following parameter definition:

$$\tau \equiv u - t \tag{10.2.2-7}$$

where

τ = Time difference parameter that will soon be identified as the correlation time for the p(t) ensemble random process.

Using (10.2.2-7) and its converse $u = t + \tau$, the expected value of (10.2.2-5) for (10.2.2-6) is:

$$\mathcal{E}\left[\frac{1}{2}\left(a_0^2 + b_0^2\right)\right] = \frac{1}{2\,T^2}\int_{u=-0.5\,T}^{0.5\,T}\int_{t=-0.5\,T}^{0.5\,T} \varphi_{pp}(t,\tau)\ dt\ du$$

$$\mathcal{E}\left[\frac{1}{2}\left(a_i^2 + b_i^2\right)\right] = \frac{2}{T^2}\int_{u=-0.5\,T}^{0.5\,T}\int_{t=-0.5\,T}^{0.5\,T} \varphi_{pp}(t,\tau)\cos\left(i\,\Delta\omega\,\tau\right) dt\ du \qquad \text{For } i > 0 \tag{10.2.2-8}$$

in which

$$\varphi_{pp}(t,\tau) \equiv \mathcal{E}\left(p(t)\ p(t+\tau)\right) \tag{10.2.2-9}$$

where

$\varphi_{pp}(t,\tau)$ = Autocorrelation function for p(t) with the separation time parameter τ identified as the correlation time.

For our purposes we will only be considering "stationary" random processes for which the autocorrelation function is independent of t and a function only of the correlation time:

$$\varphi_{pp}(t,\tau) = \varphi_{pp}(\tau) \tag{10.2.2-10}$$

Equations (10.2.2-8) can be converted into a single integration process if we perform it in a coordinate frame (u, t space) that is rotated from u, t by 45 degrees. Then the following transformation equations apply:

$$u' = \frac{1}{\sqrt{2}}(u - t) \qquad\qquad t' = \frac{1}{\sqrt{2}}(u + t) \qquad\qquad (10.2.2\text{-}11)$$

where

u', t' = Transformed u, t parameters.

Figure 10.2.2-1 depicts the Equation (10.2.2-11) axes and the integration limit boundaries associated with Equations (10.2.2-8).

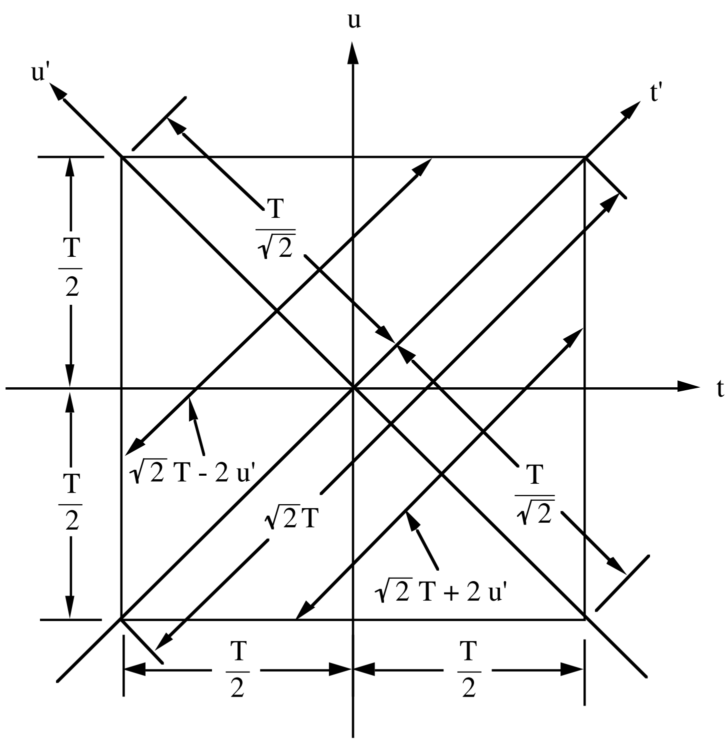


Figure 10.2.2-1 Integration Parameter Transformation

Given (10.2.2-10), the integrands in (10.2.2-8) are only a function of τ. From (10.2.2-11) and (10.2.2-7) we see that τ is constant for a particular u'. Hence, the integrands in (10.2.2-8) are also constant for a given u'. This characteristic allows the (10.2.2-8) double integration process to be simplified by performing the integration in u', t' space over the same area as the (10.2.2-8) u, t area (i.e., the area within the square having u, t boundaries of $\pm \frac{T}{2}$). Let us execute the (10.2.2-8) double integration in u', t' space first along t' (for which u' is constant), and then along u'. As can be seen from Figure 10.2.2-1, the t' integration range is from $-\frac{1}{2}\left(\sqrt{2}\,T - 2\,u'\right)$ to $+\frac{1}{2}\left(\sqrt{2}\,T - 2\,u'\right)$, or from $-\frac{1}{2}\left(\sqrt{2}\,T + 2\,u'\right)$ to $+\frac{1}{2}\left(\sqrt{2}\,T + 2\,u'\right)$, depending on whether u' is positive or negative. The u' integration range is from $-\,T/\sqrt{2}$ to $+\,T/\sqrt{2}$. Then the double integrals in (10.2.2-8) have the form:

$$\int_{u=-0.5\,T}^{0.5\,T}\int_{t=-0.5\,T}^{0.5\,T} f(\tau)\,dt\,du \;=$$

$$\int_{u'=-\frac{T}{\sqrt{2}}}^{0}\int_{t'=-\frac{1}{2}\left(\sqrt{2}\,T+2\,u'\right)}^{\frac{1}{2}\left(\sqrt{2}\,T+2\,u'\right)} f(\tau)\,dt'\,du' \;+\; \int_{u'=0}^{\frac{T}{\sqrt{2}}}\int_{t'=-\frac{1}{2}\left(\sqrt{2}\,T-2\,u'\right)}^{\frac{1}{2}\left(\sqrt{2}\,T-2\,u'\right)} f(\tau)\,dt'\,du' \qquad (10.2.2\text{-}12)$$

$$= \int_{-\frac{T}{\sqrt{2}}}^{0}\left(\sqrt{2}\,T + 2\,u'\right) f(\tau)\,du' + \int_{0}^{\frac{T}{\sqrt{2}}}\left(\sqrt{2}\,T - 2\,u'\right) f(\tau)\,du'$$

where

$f(\tau)$ = Integrand in either of the Equation (10.2.2-8) expressions.

Let us now combine (10.2.2-7) and u' from (10.2.2-11) giving:

$$u' = \frac{1}{\sqrt{2}}\,\tau \qquad\qquad du' = \frac{1}{\sqrt{2}}\,d\tau \qquad (10.2.2\text{-}13)$$

From Equation (10.2.2-13) we see that $\tau = \sqrt{2}\,u'$, hence, τ goes from - T to + T as u' goes from $-\,T/\sqrt{2}$ to $+\,T/\sqrt{2}$. Then (10.2.2-12) with (10.2.2-13) simplifies to:

$$\begin{aligned}\int_{u=-0.5\,T}^{0.5\,T}\int_{t=-0.5\,T}^{0.5\,T} f(\tau)\,dt\,du \;&=\; \int_{-T}^{0}\left(T+\tau\right) f(\tau)\,d\tau + \int_{0}^{T}\left(T-\tau\right) f(\tau)\,d\tau \\ &= \int_{-T}^{0}\left(T\,f(\tau) + \tau\,f(\tau)\right) d\tau + \int_{0}^{T}\left(T\,f(\tau) - \tau\,f(\tau)\right) d\tau\end{aligned} \qquad (10.2.2\text{-}14)$$

Equation (10.2.2-14) can be further simplified if we now consider processes of long time duration for which T is large. The $f(\tau)$ function in (10.2.2-14) is, from (10.2.2-8) and (10.2.2-10), either $\varphi_{pp}(\tau)$ or $\varphi_{pp}(\tau)\cos\left(i\,\Delta\omega\,\tau\right)$, hence has magnitude on the order of $\varphi_{pp}(\tau)$. As usual for random processes, we will assume a zero mean for the p(t) process at any time point t. We also know from the nature of random processes that $\varphi_{pp}(\tau)$ becomes small as τ becomes large. For large T, therefore, we are justified in neglecting the $\tau\, f(\tau)$ terms in (10.2.2-14) compared to the $T\, f(\tau)$ terms. Thus, for large T, (10.2.2-14) reduces to:

$$\int_{-0.5\,T}^{0.5\,T} \int_{-0.5\,T}^{0.5\,T} f(\tau)\, du\, dt \;=\; T \int_{-T}^{T} f(\tau)\, d\tau \qquad \text{For Large T} \qquad (10.2.2\text{-}15)$$

Applying (10.2.2-15) in (10.2.2-8) then gives:

For Large T:

$$\begin{aligned} \mathcal{E}\left[\frac{1}{2}\left(a_0^2 + b_0^2\right)\right] &= \frac{1}{2\,T}\int_{-T}^{T} \varphi_{pp}(\tau)\, d\tau \\ \mathcal{E}\left[\frac{1}{2}\left(a_i^2 + b_i^2\right)\right] &= \frac{2}{T}\int_{-T}^{T} \varphi_{pp}(\tau)\cos\left(i\,\Delta\omega\,\tau\right) d\tau \qquad \text{For } i > 0 \end{aligned} \qquad (10.2.2\text{-}16)$$

It is advantageous to recognize that (10.2.2-10) implies that $\varphi_{pp}(\tau)$ is a symmetrical function of τ (i.e., same value for positive or negative τ) as is easily shown by expansion of (10.2.2-9) with t equal to any time s, and then setting s to $t - \tau$:

$$\begin{aligned} \varphi_{pp}(t,\tau) &= \varphi_{pp}(\tau) = \varphi_{pp}(s,\tau) = \mathcal{E}\left(p(s)\, p(s+\tau)\right) \\ &= \mathcal{E}\left(p(t-\tau)\, p(t)\right) = \mathcal{E}\left(p(t)\, p(t-\tau)\right) = \varphi_{pp}\left(t, -\tau\right) = \varphi_{pp}(-\tau) \end{aligned} \qquad (10.2.2\text{-}17)$$

We also know that $\cos\left(i\,\Delta\omega\,\tau\right)$ in (10.2.2-16) is a symmetrical function of τ. Then, using (10.2.2-1) for $\Delta\omega$, we see that (10.2.2-16) is:

$$\begin{aligned} \mathcal{E}\left[\frac{1}{2}\left(a_0^2 + b_0^2\right)\right] &= \Delta\omega\,\frac{1}{4\,\pi}\int_{-T}^{T} \varphi_{pp}(\tau)\, d\tau = \Delta\omega\,\frac{1}{2\,\pi}\int_{0}^{T} \varphi_{pp}(\tau)\, d\tau \\ \mathcal{E}\left[\frac{1}{2}\left(a_i^2 + b_i^2\right)\right] &= \Delta\omega\,\frac{1}{\pi}\int_{-T}^{T} \varphi_{pp}(\tau)\cos\left(i\,\Delta\omega\,\tau\right) d\tau \\ &= \Delta\omega\,\frac{2}{\pi}\int_{0}^{T} \varphi_{pp}(\tau)\cos\left(i\,\Delta\omega\,\tau\right) d\tau \qquad \text{For } i > 0 \end{aligned} \qquad (10.2.2\text{-}18)$$

Now we are ready to let T go to infinity in the limit. From Equation (10.2.2-1) we see that this corresponds to letting $\Delta\omega$ go to an infinitesimal $d\omega$. We also define:

$$\omega \equiv i\,\Delta\omega \tag{10.2.2-19}$$

for which (10.2.2-18) then becomes the differential:

$$\begin{aligned} \mathcal{E}\left[\frac{1}{2}\left(a(0)^2 + b(0)^2\right)\right] &= d\omega\,\frac{1}{4}\,G(0) \\ \mathcal{E}\left[\frac{1}{2}\left(a(\omega)^2 + b(\omega)^2\right)\right] &= d\omega\,G(\omega) \qquad \text{For } \omega > 0 \end{aligned} \tag{10.2.2-20}$$

in which we have defined:

$$G(\omega) \equiv \frac{2}{\pi}\int_0^\infty \varphi_{pp}(\tau)\cos\omega\tau\,d\tau \tag{10.2.2-21}$$

where

$G(\omega)$ = Power spectral density of the p(t) random process. As an aside, it can be readily verified that (10.2.2-21) is equivalent to the Fourier transform of the autocorrelation function $\varphi_{pp}(\tau)$.

Note that in light of (10.2.2-19), the functional dependence of a_i, b_i on the i index has been modified in (10.2.2-20) to a functional dependence on the new frequency parameter ω.

For typical random processes in which $\varphi_{pp}(\tau)$ decreases to zero with increasing τ, Equation (10.2.2-21) shows that G(0) will be finite. Then the $\mathcal{E}\left[\frac{1}{2}\left(a(0)^2 + b(0)^2\right)\right]$ term in (10.2.2-20) becomes negligible, and we can write in general, without ω specificity:

$$\frac{1}{d\omega}\,\mathcal{E}\left[\frac{1}{2}\left(a(\omega)^2 + b(\omega)^2\right)\right] = G(\omega) \tag{10.2.2-22}$$

Equation (10.2.2-22) with (10.2.2-1) for p(t) will form the basis for analyses in Section 10.4 that translate Section 10.1 results into the expected system response under random vibration inputs.

Finally, let us return to (10.2.2-4) for $\mathcal{E}\left(p(t)^2\right)$ and look at this equation in the limit as T goes to infinity and $\Delta\omega$ goes to $d\omega$:

$$\mathcal{E}\left(\overline{p(t)^2}\right) = \sum_{i=0}^{\infty} \mathcal{E}\left[\frac{1}{2}\left(a_i^2 + b_i^2\right)\right] = \sum_{i=0}^{\infty} \frac{1}{\Delta\omega}\,\mathcal{E}\left[\frac{1}{2}\left(a(\omega)^2 + b(\omega)^2\right)\right]\Delta\omega$$
$$= \int_0^{\infty} \frac{1}{d\omega}\,\mathcal{E}\left[\frac{1}{2}\left(a(\omega)^2 + b(\omega)^2\right)\right] d\omega \tag{10.2.2-23}$$

Thus, with (10.2.2-22), we have the well known result:

$$\mathcal{E}\left(\overline{p(t)^2}\right) = \int_0^{\infty} G(\omega)\, d\omega \tag{10.2.2-24}$$

10.3 RESPONSE TO SINUSOIDAL SYSTEM VIBRATION INPUT

Section 10.1 analyzed the effect of sinusoidal angular rate and linear acceleration vibrations at discrete frequencies on strapdown INS computational performance. The results were summarized in Equations (10.1.5-1). Equations (10.1.5-1) are based on assumed inertial sensor sinusoidal output characteristics (amplitude and phase). In this section we will use the results of Section 10.2.1 to rewrite Equations (10.1.5-1) based on the sinusoidal output response of the inertial sensors to a specified system level input sinusoidal vibration at a particular frequency and phase angle:

$$p_{Vib}(t) = p_{Vib_0} \sin\left(\Omega t + \psi_{pVib}\right) \tag{10.3-1}$$

where

$p_{Vib}(t)$ = Generalized system sinusoidal vibration input.

Ω, p_{Vib_0}, ψ_{pVib} = Frequency, amplitude and phase angle associated with $p_{Vib}(t)$.

The $p_{Vib}(t)$ vibration could be an acceleration or angular rate vibration that is specified at some particular location and direction in the vehicle in which the INS is mounted. An example would be a sinusoidal vibration specified as an input to the INS mount along a particular INS axis. Because the INS inertial sensors are typically not rigidly connected to the mount (e.g., due to elastomeric isolators - See Section 10.5), the vibration felt by the sensors will be shifted in amplitude and phase from the $p_{Vib}(t)$ amplitude/phase. For this section, $p_{Vib}(t)$ will be considered as acting along a constant direction in the B Frame. Let us now consider strapdown inertial sensor angular and linear response in the B Frame to the general $p_{Vib}(t)$ vibration input. Using Equation (10.2.1-26) we can write for the (10.3-1) form of $p_{Vib}(t)$:

$$\underline{\theta}(t) = \underline{u}_x\, \theta_x(t) + \underline{u}_y\, \theta_y(t) + \underline{u}_z\, \theta_z(t)$$

$$\begin{aligned} \theta_x(t) &= B_{\theta_x}\, p_{Vib_0} \sin\left(\Omega t + \psi_{pVib} + \phi_{\theta_x}\right) \\ \theta_y(t) &= B_{\theta_y}\, p_{Vib_0} \sin\left(\Omega t + \psi_{pVib} + \phi_{\theta_y}\right) \\ \theta_z(t) &= B_{\theta_z}\, p_{Vib_0} \sin\left(\Omega t + \psi_{pVib} + \phi_{\theta_z}\right) \end{aligned} \tag{10.3-2}$$

$$\underline{a}_{SF}(t) = \underline{u}_x\, a_{SF_x}(t) + \underline{u}_y\, a_{SF_y}(t) + \underline{u}_z\, a_{SF_z}(t)$$

$$\begin{aligned} a_{SF_x}(t) &= B_{aSF_x}\, p_{Vib_0} \sin\left(\Omega\, t + \psi_{pVib} + \phi_{aSF_x}\right) \\ a_{SF_y}(t) &= B_{aSF_y}\, p_{Vib_0} \sin\left(\Omega\, t + \psi_{pVib} + \phi_{aSF_y}\right) \\ a_{SF_z}(t) &= B_{aSF_z}\, p_{Vib_0} \sin\left(\Omega\, t + \psi_{pVib} + \phi_{aSF_z}\right) \end{aligned} \tag{10.3-3}$$

where

$\underline{\theta}(t)$ = B Frame vibration "angle" vector which we define as the integrated B Frame angular rate. We also define $\underline{\theta}(t)$ such that the angular rate in the B Frame (sensed by the strapdown angular rate sensors) is the derivative of $\underline{\theta}(t)$.

$\underline{a}_{SF}(t)$ = B Frame specific force acceleration vector that would be measured by the strapdown accelerometers.

$\theta_x(t),\ \theta_y(t),\ \theta_z(t),\ a_{SF_x}(t),\ a_{SF_y}(t),\ a_{SF_z}(t)$ = B Frame X, Y, Z components of $\underline{\theta}(t)$ and $\underline{a}_{SF}(t)$.

$B_{\theta_x}, B_{\theta_y}, B_{\theta_z}, \phi_{\theta_x}, \phi_{\theta_y}, \phi_{\theta_z}$ = Amplitude ratios and phase angles associated with the transfer functions relating $p_{Vib}(t)$ to the $\theta_x(t)$, $\theta_y(t)$, $\theta_z(t)$ response at frequency Ω.

$B_{a_{SF_x}}, B_{a_{SF_y}}, B_{a_{SF_z}}, \phi_{a_{SF_x}}, \phi_{a_{SF_y}}, \phi_{a_{SF_z}}$ = Amplitude ratios and phase angles associated with the transfer functions relating $p_{Vib}(t)$ to the $a_{SF_x}(t)$, $a_{SF_y}(t)$, $a_{SF_z}(t)$ response at frequency Ω.

A more general treatment would consider $p_{Vib}(t)$ as a three component vector with each component being a sinusoid with unique phase relationship between the component sinusoids. The effect on $\underline{\theta}(t)$ and $\underline{a}_{SF}(t)$ in Equations (10.3-2) - (10.3-3) would be to linearly add terms for the additional axis components, of identical form to those shown. This general treatment adds a considerable degree of complexity, and doesn't alter the essential characteristic of results obtained. For simplicity then, we will only consider the single axis $p_{Vib}(t)$ definition. It is worthy of note to point out that typical vibration test specifications for INS's are also based on the vibration input being along a fixed axis relative to the INS, hence, the single axis $p_{Vib}(t)$ approach has merit if one is concerned with performance prediction for a vibration test.

Let us now consider how $p_{Vib}(t)$ and its (10.3-2) - (10.3-3) response impacts the $\dot{\underline{\beta}}_m$, $\Delta\dot{\underline{v}}_{Scul_m}$ and $\delta\dot{\underline{v}}_{SFScul_{SnsDyn}}$ performance parameters in Equations (10.1.5-1). The results will then be easily extended to the remaining (10.1.5-1) vibration performance parameters.

From the Section 10.1.1.2.1 analyses leading to $\dot{\underline{\beta}}_m$ (in Equation (10.1.1.2.1-14)), we know that the $\dot{\underline{\beta}}_m$ equation is based on the following inertial sensor outputs (from (10.1.1-1)):

$$\underline{\theta}(t) = \underline{u}_x \, \theta_x(t) + \underline{u}_y \, \theta_y(t)$$
$$\theta_x(t) = \theta_{0_x} \sin\left(\Omega\, t - \varphi_{\theta_x}\right) \qquad \theta_y(t) = \theta_{0_y} \sin\left(\Omega\, t - \varphi_{\theta_y}\right) \tag{10.3-4}$$

The Z component response of $\dot{\underline{\beta}}_m$ to the previous input is from (10.1.1.2.1-14):

$$\dot{\beta}_{m_z} = \frac{1}{2}\,\Omega\,\theta_{0_x}\,\theta_{0_y} \sin\left(\varphi_{\theta_y} - \varphi_{\theta_x}\right)\left(1 - \frac{\sin\,\Omega T_m}{\Omega T_m}\right) \tag{10.3-5}$$

where

$\dot{\beta}_{m_z}$ = B Frame Z component of $\dot{\underline{\beta}}_m$.

If the derivation of $\dot{\underline{\beta}}_m$ in Section 10.1.1.2.1 is reviewed, it should be clear that each component of $\dot{\underline{\beta}}_m$ is formed from the products of angular vibration terms along the other component axes. Hence, products of X/Y vibrations produced the $\dot{\beta}_{m_z}$ Z component of $\dot{\underline{\beta}}_m$, and Y/Z, Z/X vibration products would have produced X and Y components for $\dot{\underline{\beta}}_m$. We should add that this same general rule applies for all of the vibration performance vector parameters in summary Equations (10.1.5-1). Thus, had a $\theta_z(t)$ component been included in (10.3-4), the (10.3-5) $\dot{\beta}_{z_m}$ result would have been the same, but $\dot{\beta}_{m_x}$, $\dot{\beta}_{m_y}$ components would also have been present. Values for these $\dot{\beta}_{m_x}$, $\dot{\beta}_{m_y}$ can be readily derived directly from (10.3-5) by permuting subscripts. With this general understanding, we will continue this section dealing only with the Z axis response of the (10.1.5-1) performance parameters as representative of each B Frame axis response to a B Frame three-axis vibration.

Continuing with the $\dot{\beta}_{m_Z}$ analysis, then, we quickly see by comparing the X, Y components of (10.3-4) and (10.3-2) that:

$$
\begin{aligned}
\theta_{0_x} &= B_{\theta_x}\, p_{Vib_0} \qquad & \varphi_{\theta_x} &= -\,\psi_{pVib} - \phi_{\theta_x} \\
\theta_{0_y} &= B_{\theta_y}\, p_{Vib_0} \qquad & \varphi_{\theta_y} &= -\,\psi_{pVib} - \phi_{\theta_y}
\end{aligned}
\tag{10.3-6}
$$

Substituting (10.3-6) into (10.3-5) provides the desired equation for $\dot{\beta}_{m_Z}$ as a function of $p_{Vib}(t)$ characteristics.

$$
\dot{\beta}_{m_Z} = \frac{1}{2}\,\Omega\, B_{\theta_x}\, B_{\theta_y}\, p_{Vib_0}^2 \sin\left(\phi_{\theta_x} - \phi_{\theta_y}\right)\left(1 - \frac{\sin\,\Omega T_m}{\Omega T_m}\right) \tag{10.3-7}
$$

Let's apply a similar treatment for the $\Delta\dot{\underline{v}}_{Scul_m}$ term in (10.1.5-1). From the Section 10.1.2.2.1 analyses leading to $\Delta\dot{\underline{v}}_{Scul_m}$ (in Equation (10.1.2.2.1-11)), we know that the $\Delta\dot{\underline{v}}_{Scul_m}$ equation is based on the (10.1.2-1) and (10.1.2-3) inertial sensor inputs having angular motion about B Frame axis X and linear motion along B Frame axis Y producing $\Delta\dot{\underline{v}}_{Scul_m}$ along B Frame axis Z. From the same analyses, it should also be apparent that sinusoidal angular motion about B Frame Y coupled with linear motion along B Frame X at the same frequency will also produce $\Delta\dot{\underline{v}}_{Scul_m}$ along B Frame Z. A more general version of the (10.1.5-1) $\Delta\dot{\underline{v}}_{Scul_m}$ expression that accounts for combined B Frame X, Y angular/linear vibration is obtained by inspection and direct expansion of (10.1.2-1), (10.1.2-3) and $\Delta\dot{\underline{v}}_{Scul_m}$ in (10.1.5-1) using subscript permutation for the Y-angular/X-linear vibration portion:

$$
\begin{aligned}
&\underline{\theta}(t) = \underline{u}_x\, \theta_x(t) + \underline{u}_y\, \theta_y(t) \\
&\theta_x(t) = \theta_{0_x} \sin\left(\Omega\, t - \varphi_{\theta_x}\right) \qquad \theta_y(t) = \theta_{0_y} \sin\left(\Omega\, t - \varphi_{\theta_y}\right) \\
&\underline{a}_{SF}(t) = \underline{u}_x\, a_{SF_x}(t) + \underline{u}_y\, a_{SF_y}(t) \\
&a_{SF_x}(t) = a_{SF0_x} \sin\left(\Omega\, t - \varphi_{aSF_x}\right) \qquad a_{SF_y}(t) = a_{SF0_y} \sin\left(\Omega\, t - \varphi_{aSF_y}\right)
\end{aligned}
\tag{10.3-8}
$$

$$
\begin{aligned}
\Delta\dot{v}_{Scul\text{-}m_Z} = \frac{1}{2}\Big[&\theta_{0_x}\, a_{SF0_y} \cos\left(\varphi_{aSF_y} - \varphi_{\theta_x}\right) \\
&- \theta_{0_y}\, a_{SF0_x} \cos\left(\varphi_{aSF_x} - \varphi_{\theta_y}\right)\Big]\left(1 - \frac{\sin\,\Omega T_m}{\Omega T_m}\right)
\end{aligned}
\tag{10.3-9}
$$

where

$$\Delta\dot{v}_{Scul\text{-}m_z} = \text{B Frame Z component of } \Delta\underline{\dot{v}}_{Scul_m}.$$

Comparing the X, Y components of (10.3-8) and (10.3-2) - (10.3-3) we quickly see that:

$$\begin{aligned}
\theta_{0_x} &= B_{\theta_x}\, p_{Vib_0} & \varphi_{\theta_x} &= -\,\psi_{pVib} - \phi_{\theta_x} \\
\theta_{0_y} &= B_{\theta_y}\, p_{Vib_0} & \varphi_{\theta_y} &= -\,\psi_{pVib} - \phi_{\theta_y} \\
a_{SF0_x} &= B_{a_{SF_x}}\, p_{Vib_0} & \varphi_{a_{SF_x}} &= -\,\psi_{pVib} - \phi_{a_{SF_x}} \\
a_{SF0_y} &= B_{a_{SF_y}}\, p_{Vib_0} & \varphi_{a_{SF_y}} &= -\,\psi_{pVib} - \phi_{a_{SF_y}}
\end{aligned} \qquad (10.3\text{-}10)$$

Substituting (10.3-10) into the (10.3-9) $\Delta\dot{v}_{Scul\text{-}m_z}$ expression provides the desired equation for $\Delta\dot{v}_{Scul\text{-}m_z}$ as a function of $p_{Vib}(t)$ characteristics:

$$\begin{aligned}
\Delta\dot{v}_{Scul\text{-}m_z} = \frac{1}{2}\, p_{Vib_0}^2 \Big[& B_{\theta_x} B_{a_{SF_y}} \cos\left(\phi_{\theta_x} - \phi_{a_{SF_y}}\right) \\
& - B_{\theta_y} B_{a_{SF_x}} \cos\left(\phi_{\theta_y} - \phi_{a_{SF_x}}\right)\Big] \left(1 - \frac{\sin \Omega T_m}{\Omega T_m}\right)
\end{aligned} \qquad (10.3\text{-}11)$$

For the $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$ performance parameter in (10.1.5-1) we must account for the dynamic response of the inertial sensors to inputs derived from $p_{Vib}(t)$. From the Section 10.1.4.2 analyses leading to $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$ (in Equation (10.1.4.2-6)), we know that the $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$ equation is based on (10.1.4.2-1) - (10.1.4.2-2) inertial sensor inputs/outputs having angular motion about B Frame axis X and linear motion along B Frame axis Y producing $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$ along B Frame axis Z. From the same analyses, it should also be apparent that sinusoidal angular motion about B Frame Y coupled with linear motion along B Frame X at the same frequency will also produce $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$ along B Frame Z. A more general version of the (10.1.5-1) $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$ expression that accounts for combined B Frame X, Y angular/linear vibration is obtained by inspection and direct expansion of (10.1.4.2-1), (10.1.4.2-2) and $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$ in (10.1.5-1) using subscript permutation for the Y-angular/X-linear vibration portion:

$$\begin{aligned}
\underline{\omega}_{IB}(t) &= \underline{u}_x\, \theta_{0_x}\, \Omega \cos\left(\Omega\, t - \varphi_{\theta_x}\right) + \underline{u}_y\, \theta_{0_y}\, \Omega \cos\left(\Omega\, t - \varphi_{\theta_y}\right) \\
\underline{a}_{SF}(t) &= \underline{u}_x\, a_{SF0_x} \sin\left(\Omega\, t - \varphi_{a_{SF_x}}\right) + \underline{u}_y\, a_{SF0_y} \sin\left(\Omega\, t - \varphi_{a_{SF_y}}\right)
\end{aligned} \qquad (10.3\text{-}12)$$

$$\begin{aligned}
\underline{\omega}_{ARS}(t) &= \underline{u}_x\, \omega_{ARS0_x} \cos\left(\Omega\, t - \varphi_{\omega_{ARS_x}}\right) + \underline{u}_y\, \omega_{ARS0_y} \cos\left(\Omega\, t - \varphi_{\omega_{ARS_y}}\right) \\
\underline{a}_{Accl}(t) &= \underline{u}_x\, a_{Accl0_x} \sin\left(\Omega\, t - \varphi_{a_{Accl\,x}}\right) + \underline{u}_y\, a_{Accl0_y} \sin\left(\Omega\, t - \varphi_{a_{Accl\,y}}\right)
\end{aligned} \qquad (10.3\text{-}13)$$

$$\begin{aligned}
\delta \dot{v}_{SF/Scul/SnsDyn_z} = \frac{1}{2}\Bigg[& \omega_{ARS0_x} \frac{1}{\Omega}\, a_{Accl0_y} \cos\left(\varphi_{a_{Accl_y}} - \varphi_{\omega_{ARS_x}}\right) \\
& - \theta_{0_x}\, a_{SF0_y} \cos\left(\varphi_{a_{SF_y}} - \varphi_{\theta_x}\right)\Bigg] - \frac{1}{2}\Bigg[\omega_{ARS0_y} \frac{1}{\Omega}\, a_{Accl0_x} \cos\left(\varphi_{a_{Accl\,x}} - \varphi_{\omega_{ARS_y}}\right) \\
& - \theta_{0_y}\, a_{SF0_x} \cos\left(\varphi_{a_{SF_x}} - \varphi_{\theta_y}\right)\Bigg]
\end{aligned} \qquad (10.3\text{-}14)$$

where

$\delta \dot{v}_{SF/Scul/SnsDyn_z}$ = B Frame Z component of $\delta \underline{\dot{v}}_{SF/Scul_{SnsDyn}}$.

$\omega_{ARS0_x}, \omega_{ARS0_y}, a_{Accl0_x}, a_{Accl0_y}$ = Amplitudes for the $\underline{\omega}_{ARS}(t)$, $\underline{a}_{Accl}(t)$ angular rate sensor and accelerometer B Frame X, Y axis sinusoidal outputs at frequency Ω.

$\varphi_{\omega_{ARS_x}}, \varphi_{\omega_{ARS_y}}, \varphi_{a_{Accl\,x}}, \varphi_{a_{Accl\,y}}$ = Phase angles for the $\underline{\omega}_{ARS}(t)$, $\underline{a}_{Accl}(t)$ angular rate sensor and accelerometer B Frame X, Y axis sinusoidal outputs at frequency Ω.

The $\underline{\omega}_{IB}(t)$ expression in (10.3-12) is the derivative of the equivalent angular vibration equation:

$$\underline{\theta}(t) = \underline{u}_x\, \theta_{0_x} \sin\left(\Omega\, t - \varphi_{\theta_x}\right) + \underline{u}_y\, \theta_{0_y} \sin\left(\Omega\, t - \varphi_{\theta_y}\right) \qquad (10.3\text{-}15)$$

Comparing (10.3-15) and the $\underline{a}_{SF}(t)$ expression in (10.3-12) with the X, Y components of (10.3-2) - (10.3-3) we see that (10.3-10) still applies. Equation (10.2.1-26) shows that the dependence of $\underline{\omega}_{ARS}(t)$, $\underline{a}_{Accl}(t)$ on $\underline{\omega}_{IB}(t)$, $\underline{a}_{SF}(t)$ as defined in (10.3-12) - (10.3-13) is:

$$\begin{aligned}
\underline{\omega}_{ARS}(t) &= \underline{u}_x\, B_{\omega_{ARS_x}}\, \theta_{0_x}\, \Omega \cos\left(\Omega\, t - \varphi_{\theta_x} + \phi_{\omega_{ARS_x}}\right) \\
&\quad + \underline{u}_y\, B_{\omega_{ARS_y}}\, \theta_{0_y}\, \Omega \cos\left(\Omega\, t - \varphi_{\theta_y} + \phi_{\omega_{ARS_y}}\right) \\
\underline{a}_{Accl}(t) &= \underline{u}_x\, B_{aAccl_x}\, a_{SF0_x} \sin\left(\Omega\, t - \varphi_{a_{SF_x}} + \phi_{aAccl_x}\right) \\
&\quad + \underline{u}_y\, B_{aAccl_y}\, a_{SF0_y} \sin\left(\Omega\, t - \varphi_{a_{SF_y}} + \phi_{aAccl_y}\right)
\end{aligned} \qquad (10.3\text{-}16)$$

where

$B_{\omega ARS_x}, B_{\omega ARS_y}, \phi_{\omega ARS_x}, \phi_{\omega ARS_y}$ = Amplitude ratios and phase angles associated with the X, Y angular rate sensor response to applied inputs at frequency Ω.

$B_{aAccl_x}, B_{aAccl_y}, \phi_{aAccl_x}, \phi_{aAccl_y}$ = Amplitude ratios and phase angles associated with the X, Y accelerometer response to applied inputs at frequency Ω.

Comparing (10.3-16) and (10.3-13) we see that:

$$
\begin{aligned}
\omega_{ARS0_x} &= B_{\omega ARS_x}\,\theta_{0_x}\,\Omega & \varphi_{\omega ARS_x} &= \varphi_{\theta_x} - \phi_{\omega ARS_x} \\
\omega_{ARS0_y} &= B_{\omega ARS_y}\,\theta_{0_y}\,\Omega & \varphi_{\omega ARS_y} &= \varphi_{\theta_y} - \phi_{\omega ARS_y} \\
a_{Accl0_x} &= B_{aAccl_x}\,a_{SF0_x} & \varphi_{aAccl_x} &= \varphi_{aSF_x} - \phi_{aAccl_x} \\
a_{Accl0_y} &= B_{aAccl_y}\,a_{SF0_y} & \varphi_{aAccl_y} &= \varphi_{aSF_y} - \phi_{aAccl_y}
\end{aligned}
\tag{10.3-17}
$$

Combining (10.3-17) and (10.3-10) then finds:

$$
\begin{aligned}
\omega_{ARS0_x} &= B_{\omega ARS_x}\,B_{\theta_x}\,p_{Vib_0}\,\Omega & \varphi_{\omega ARS_x} &= -\,\psi_{pVib} - \phi_{\theta_x} - \phi_{\omega ARS_x} \\
\omega_{ARS0_y} &= B_{\omega ARS_y}\,B_{\theta_y}\,p_{Vib_0}\,\Omega & \varphi_{\omega ARS_y} &= -\,\psi_{pVib} - \phi_{\theta_y} - \phi_{\omega ARS_y} \\
a_{Accl0_x} &= B_{aAccl_x}\,B_{aSF_x}\,p_{Vib_0} & \varphi_{aAccl_x} &= -\,\psi_{pVib} - \phi_{aSF_x} - \phi_{aAccl_x} \\
a_{Accl0_y} &= B_{aAccl_y}\,B_{aSF_y}\,p_{Vib_0} & \varphi_{aAccl_y} &= -\,\psi_{pVib} - \phi_{aSF_y} - \phi_{aAccl_y}
\end{aligned}
\tag{10.3-18}
$$

Finally, we substitute (10.3-18) and (10.3-10) into the (10.3-14) $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ expression to obtain the equation for $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ as a function of $p_{Vib}(t)$ characteristics:

$$
\begin{aligned}
\delta\dot{v}_{SF/Scul/SnsDyn_z} = \frac{1}{2}\,p_{Vib_0}^2 \Big\{ & B_{\theta_x}\,B_{aSF_y}\Big[B_{\omega ARS_x}\,B_{aAccl_y}\cos\Big(\phi_{\theta_x} \\
& + \phi_{\omega ARS_x} - \phi_{aSF_y} - \phi_{aAccl_y}\Big) - \cos\Big(\phi_{\theta_x} - \phi_{aSF_y}\Big)\Big] \\
& - B_{\theta_y}\,B_{aSF_x}\Big[B_{\omega ARS_y}\,B_{aAccl_x}\cos\Big(\phi_{\theta_y} + \phi_{\omega ARS_y} - \phi_{aSF_x} - \phi_{aAccl_x}\Big) - \cos\Big(\phi_{\theta_y} - \phi_{aSF_x}\Big)\Big]\Big\}
\end{aligned}
\tag{10.3-19}
$$

The above treatment can also be applied to the remaining performance parameters in Equations (10.1.5-1). By inspection of Equation (10.3-11) compared to (10.3-9) for

$\Delta\dot{v}_{Scul\text{-}m_z}$, and Equation (10.3-7) compared to (10.3-5) for $\dot{\beta}_{m_z}$, the equivalent results for the remaining (10.1.5-1) parameters should be obvious. The overall result (including the previously derived $\Delta\dot{v}_{Scul\text{-}m_z}$, $\dot{\beta}_{m_z}$, $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ equations) is given by:

$$\dot{\Phi}_{Con_z} = \frac{1}{2}\,\Omega\,B_{\theta_x}\,B_{\theta_y}\,p^2_{Vib_0}\,\sin\left(\phi_{\theta_x} - \phi_{\theta_y}\right)$$

$$\dot{\beta}_{m_z} = \frac{1}{2}\,\Omega\,B_{\theta_x}\,B_{\theta_y}\,p^2_{Vib_0}\,\sin\left(\phi_{\theta_x} - \phi_{\theta_y}\right)\left(1 - \frac{\sin\,\Omega T_m}{\Omega T_m}\right)$$

$$\dot{\beta}_{Algo\text{-}m_z} = \frac{1}{2}\,\Omega\,B_{\theta_x}\,B_{\theta_y}\,p^2_{Vib_0}\,\sin\left(\phi_{\theta_x} - \phi_{\theta_y}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos\,\Omega\,T_l\right)\right]\left(\frac{\sin\,\Omega\,T_l}{\Omega\,T_l}\right) - \frac{\sin\,\Omega\,T_m}{\Omega\,T_m}\right\}$$

$$\delta\dot{\Phi}_{Algo\text{-}m_z} = \delta\dot{\beta}_{Algo\text{-}m_z} = \frac{1}{2}\,\Omega\,B_{\theta_x}\,B_{\theta_y}\,p^2_{Vib_0}\,\sin\left(\phi_{\theta_x} - \phi_{\theta_y}\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos\,\Omega\,T_l\right)\right]\left(\frac{\sin\,\Omega\,T_l}{\Omega\,T_l}\right) - 1\right\}$$

$$\dot{v}_{SF/Scul_z} = \frac{1}{2}\,p^2_{Vib_0}\left[B_{\theta_x}\,B_{aSF_y}\cos\left(\phi_{\theta_x} - \phi_{aSF_y}\right) - B_{\theta_y}\,B_{aSF_x}\cos\left(\phi_{\theta_y} - \phi_{aSF_x}\right)\right] \qquad (10.3\text{-}20)$$

$$\Delta\dot{v}_{Scul\text{-}m_z} = \frac{1}{2}\,p^2_{Vib_0}\left[B_{\theta_x}\,B_{aSF_y}\cos\left(\phi_{\theta_x} - \phi_{aSF_y}\right) - B_{\theta_y}\,B_{aSF_x}\cos\left(\phi_{\theta_y} - \phi_{aSF_x}\right)\right]\left(1 - \frac{\sin\,\Omega T_m}{\Omega T_m}\right)$$

$$\Delta\dot{v}_{Scul/Algo\text{-}m_z} = \frac{1}{2}\,p^2_{Vib_0}\left[B_{\theta_x}\,B_{aSF_y}\cos\left(\phi_{\theta_x} - \phi_{aSF_y}\right) - B_{\theta_y}\,B_{aSF_x}\cos\left(\phi_{\theta_y} - \phi_{aSF_x}\right)\right]\left\{\left[1 + \frac{1}{3}\left(1 - \cos\,\Omega\,T_l\right)\right]\left(\frac{\sin\,\Omega\,T_l}{\Omega\,T_l}\right) - \frac{\sin\,\Omega\,T_m}{\Omega\,T_m}\right\}$$

(Continued)

$$\delta\dot{v}_{SF/Algo\text{-}m_z} = \delta\Delta\dot{v}_{Scul/Algo\text{-}m_z} = \frac{1}{2} p_{Vib_0}^2 \Big[B_{\theta_x} B_{a_{SF_y}} \cos\left(\phi_{\theta_x} - \phi_{a_{SF_y}}\right)$$

$$- B_{\theta_y} B_{a_{SF_x}} \cos\left(\phi_{\theta_y} - \phi_{a_{SF_x}}\right)\Big] \left\{ \left[1 + \frac{1}{3}\left(1 - \cos\,\Omega\,T_l\right)\right] \left(\frac{\sin\,\Omega\,T_l}{\Omega\,T_l}\right) - 1 \right\}$$

$$\delta\underline{R}_{SF/Algo}(t) = -\,\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,p_{Vib_0}\,B_{aSF} \Bigg\{ \Omega(t - t_0) \left(\frac{f_1(\Omega'\,T_l)}{2\,f_2(\Omega'\,T_l)} - \frac{\Omega'}{\Omega} \right.$$

$$\left. + \frac{1}{12}\,(\Omega'\,T_l)^2\,f_1(\Omega'\,T_l) \right) \Big[\cos\left(\Omega\,t_0 + \psi_{pVib} + \phi_{aSF}\right)\; f_1\big(\Omega'(t - t_0)\big)$$

$$- \sin\,(\Omega\,t_0 + \psi_{pVib} + \phi_{aSF})\;\Omega'\,(t - t_0)\;f_2\big(\Omega'(t - t_0)\big)\Big]$$

$$- \frac{1}{12}\,\Omega\,T_l \Big[\cos\left(\Omega'(t - t_0) + \Omega\,t_0 + \psi_{pVib} + \phi_{aSF}\right) \qquad \text{(10.3-20) (Continued)}$$

$$- \cos\,(\Omega\,t_0 + \psi_{pVib} + \phi_{aSF})\Big]\;(1 - \cos\,\Omega'\,T_l) \Bigg\}$$

$$\Omega' = \Omega - \frac{2\,\pi}{T_l}\left(\frac{\Omega\,T_l}{2\,\pi}\right)_{Intgr}$$

f_1 and f_2 are defined by Equations (10.1.3.2.3-16)

$$\overline{f(t)} = \frac{1}{2}\,B_{h_x}\,B_{h_y}\,p_{Vib_0}^2 \cos\left(\phi_{h_x} - \phi_{h_y}\right)$$

$$\overline{\delta a_{Accl_{G2}}} = L_{G2}\,\frac{1}{2}\,B_{Accl_{Inpt}}\,B_{Accl_{Pend}}\,p_{Vib_0}^2 \cos\left(\phi_{a_{Accl\,Inpt}} - \phi_{a_{Accl\,Pend}}\right)$$

$$\overline{\delta a_{Accl_{Aniso}}} = L_{Aniso}\,\frac{1}{2}\,B_{\omega_{IB_{Accl/Inpt}}}\,B_{\omega_{IB_{Accl/Pend}}}\,p_{Vib_0}^2 \cos\left(\phi_{\omega_{IB_{Accl/Inpt}}} - \phi_{\omega_{IB_{Accl/Pend}}}\right)$$

$$\delta\dot{v}_{SF/Scul/SnsDyn_z} = \frac{1}{2}\,p_{Vib_0}^2 \Big\{ B_{\theta_x} B_{a_{SF_y}} \Big[B_{\omega_{ARS_x}}\,B_{aAccl_y} \cos\big(\phi_{\theta_x}$$

$$+ \phi_{\omega_{ARS_x}} - \phi_{a_{SF_y}} - \phi_{aAccl_y}\big) - \cos\left(\phi_{\theta_x} - \phi_{a_{SF_y}}\right)\Big]$$

$$- B_{\theta_y} B_{a_{SF_x}} \Big[B_{\omega_{ARS_y}}\,B_{aAccl_x} \cos\left(\phi_{\theta_y} + \phi_{\omega_{ARS_y}} - \phi_{a_{SF_x}} - \phi_{aAccl_x}\right) - \cos\left(\phi_{\theta_y} - \phi_{a_{SF_x}}\right)\Big]\Big\}$$

where

$\dot{\Phi}_{Con_z}, \dot{\beta}_{Algo\text{-}m_z}, \delta\dot{\Phi}_{Algo\text{-}m_z}, \delta\dot{\beta}_{Algo\text{-}m_z}$ = B Frame Z axis components of $\dot{\underline{\Phi}}_{Con}, \dot{\underline{\beta}}_{Algo_m}, \delta\dot{\underline{\Phi}}_{Algo_m}, \delta\dot{\underline{\beta}}_{Algo_m}$.

$\dot{v}_{SF/Scul_z}, \Delta\dot{v}_{Scul/Algo\text{-}m_z}, \delta\dot{v}_{SF/Algo\text{-}m_z}, \delta\Delta\dot{v}_{Scul/Algo\text{-}m_z}$ = B Frame Z axis components of $\dot{\underline{v}}_{SF_{Scul}}, \Delta\dot{\underline{v}}_{Scul/Algo_m}, \delta\dot{\underline{v}}_{SF/Algo_m}, \delta\Delta\dot{\underline{v}}_{Scul/Algo_m}$

B_{aSF}, ϕ_{aSF} = Amplitude ratio and phase angle associated with the transfer function relating $p_{Vib}(t)$ to the response of $\delta \underline{R}_{SF/Algo}(t)$ in Equations (10.1.5-1) at frequency Ω.

B_{h_x}, B_{h_y}, ϕ_{h_x}, ϕ_{h_y} = Amplitude ratios and phase angles associated with the transfer functions relating $p_{Vib}(t)$ to the response of the Section 10.1.4.1 $h_x(t)$, $h_y(t)$ parameters at frequency Ω.

$B_{Accl_{Inpt}}$, $B_{Accl_{Pend}}$, $\phi_{a_{Accl\,Inpt}}$, $\phi_{a_{Accl\,Pend}}$ = Amplitude ratios and phase angles associated with the transfer functions relating $p_{Vib}(t)$ to the response of the Section 10.1.4.1 $a_{Accl_{Inpt}}(t)$, $a_{Accl_{Pend}}(t)$ parameters at frequency Ω.

$B_{\omega_{IB_{Accl/Inpt}}}$, $B_{\omega_{IB_{Accl/Pend}}}$, $\phi_{\omega_{IB_{Accl/Inpt}}}$, $\phi_{\omega_{IB_{Accl/Pend}}}$ = Amplitude ratios and phase angle associated with the transfer functions relating $p_{Vib}(t)$ to the response of the Section 10.1.4.1 $\omega_{IB_{Accl/Inpt}}(t)$, $\omega_{IB_{Accl/Pend}}(t)$ parameters at frequency Ω.

10.4 RESPONSE TO RANDOM SYSTEM VIBRATION INPUT

In Section 10.3 we developed analytical expressions for several INS vibration sensitive performance parameters for a general system sinusoidal vibration input p(t) at a particular frequency along a particular B frame direction. The overall results have been summarized in Equations (10.3-20). In this section we analyze the problem of defining INS vibration performance in the presence of a general random vibration system input profile. The problem will be addressed as an extension of the Section 10.3 discrete frequency results to the general random vibration profile case using the Section 10.2.2 Fourier series approach. We begin, as in Section 10.2.2, by first defining the vibration input profile as the general Fourier series:

$$
\begin{aligned}
p_{Vib}(t) &= \sum_{0}^{\infty} p_{Vib_i}(t) \\
p_{Vib_i}(t) &= a_{Vib_i} \sin \omega_i t + b_{Vib_i} \cos \omega_i t
\end{aligned}
\qquad (10.4\text{-}1)
$$

where

$p_{Vib}(t)$ = General system vibration input (angular or linear) in a fixed B Frame direction.

$p_{Vib_i}(t)$ = Contribution of frequency ω_i components to $p_{Vib}(t)$.

a_{Vib_i}, b_{Vib_i} = Fourier series coefficients associated with $p_{Vib}(t)$.

We assume, of course, as in Equations (10.2.2-19) and (10.2.2-1), that ω_i satisfies:

$$\omega_i = i\,\Delta\omega \qquad \Delta\omega = \frac{2\,\pi}{T} \tag{10.4-2}$$

where

T = Width of the time interval over which the Fourier series fits the $p_{Vib}(t)$ time function.

Equation (10.4-1) for $p_{Vib_i}(t)$ can also be written in the alternative form:

$$p_{Vib_i}(t) = p_{Vib_{0/i}} \sin\left(\omega_i t + \psi_{pVib_i}\right) \tag{10.4-3}$$

where

$p_{Vib_{0/i}}, \psi_{pVib_i}$ = Amplitude and phase angle for the $p_{Vib_i}(t)$ sinusoid.

Upon expansion we get:

$$p_{Vib_i}(t) = p_{Vib_{0/i}}\left(\sin \omega_i t \cos \psi_{pVib_i} + \cos \omega_i t \sin \psi_{pVib_i}\right) \tag{10.4-4}$$

Comparing (10.4-4) with the $p_{Vib_i}(t)$ expression in (10.4-1) we see that:

$$p_{Vib_{0/i}} \cos \psi_{pVib_i} = a_{Vib_i} \qquad p_{Vib_{0/i}} \sin \psi_{pVib_i} = b_{Vib_i} \tag{10.4-5}$$

from which:

$$p_{Vib_{0/i}} = \sqrt{a_{Vib_i}^2 + b_{Vib_i}^2} \qquad \psi_{pVib_i} = \tan^{-1}\frac{b_{Vib_i}}{a_{Vib_i}} \tag{10.4-6}$$

Section 10.4.1 to follow evaluates the $\dot{\beta}_{m_z}$ parameter in Equations (10.3-20) under the above $p_{Vib}(t)$ vibration exposure. The result is then extended as in Section 10.2.2, to the response under random vibration. Then, using the $\dot{\beta}_{m_z}$ results as a template, the random vibration response of the remaining Equation (10.3-20) parameters is determined. An exception is the (10.3-20) position algorithm error parameter which, due to the complexity of its random response development, is derived separately in Section 10.4.2.

10.4.1 ATTITUDE/VELOCITY RESPONSE TO RANDOM VIBRATION INPUT

Let us now address the problem of determining the $\dot{\beta}_{m_z}$ coning performance parameter response to random vibration. As in Section 10.2.2, we first find the response of $\dot{\beta}_{m_z}$ to the Section 10.4 defined $p_{Vib}(t)$ vibration profile. We begin by writing the equivalent to (10.3-2) for the B Frame angular response to each $p_{Vib_i}(t)$ component from the (10.4-3) definition:

$$
\begin{aligned}
\underline{\theta}_i(t) &= \underline{u}_x\, \theta_{x_i}(t) + \underline{u}_y\, \theta_{y_i}(t) + \underline{u}_z\, \theta_{z_i}(t) \\
\theta_{x_i}(t) &= B_{\theta_{x/i}}\, p_{Vib_{0/i}} \sin\left(\omega_i t + \psi_{pVib_i} + \phi_{\theta_{x/i}}\right) \\
\theta_{y_i}(t) &= B_{\theta_{y/i}}\, p_{Vib_{0/i}} \sin\left(\omega_i t + \psi_{pVib_i} + \phi_{\theta_{y/i}}\right) \\
\theta_{z_i}(t) &= B_{\theta_{z/i}}\, p_{Vib_{0/i}} \sin\left(\omega_i t + \psi_{pVib_i} + \phi_{\theta_{z/i}}\right)
\end{aligned}
\qquad (10.4.1\text{-}1)
$$

where

$\underline{\theta}_i(t)$ = B Frame vibration "angle" vector response to $p_{Vib_i}(t)$ which we also define as the integrated B Frame angular rate response.

$\theta_{x_i}(t),\ \theta_{y_i}(t),\ \theta_{z_i}(t)$ = B Frame X, Y, Z components of $\underline{\theta}_i(t)$.

$B_{\theta_{x/i}}, B_{\theta_{y/i}}, B_{\theta_{z/i}}, \phi_{\theta_{x/i}}, \phi_{\theta_{y/i}}, \phi_{\theta_{z/i}}$ = Amplitude ratios and phase angles associated with the transfer functions relating $p_{Vib_i}(t)$ to the $\theta_{x_i}(t),\ \theta_{y_i}(t),\ \theta_{z_i}(t)$ response at frequency ω_i.

Note (as in Section 10.3) that for a more general multi-axis $p_{Vib}(t)$ definition, Equations (10.4-1) - (10.4.1-1) would contain additional terms of identical form for each added $p_{Vib}(t)$ component. This section will eventually consider $p_{Vib}(t)$ to be a random process. For the general $p_{Vib}(t)$ case, if we consider each added $p_{Vib}(t)$ component as a random process that is independent of the others, the net statistical result eventually obtained will be the linear sum of the effect of each $p_{Vib}(t)$ component acting independently. Based on this understanding, we will continue the discussion (as in Section 10.3) based on our simplified single B Frame direction $p_{Vib}(t)$ component model, which is also consistent with $p_{Vib}(t)$ definitions in INS test specifications.

We assume a linear response of $\underline{\theta}_i(t)$ to $p_{Vib_i}(t)$, hence, the total angular response around axes X, Y and Z (i.e., $\theta_x(t),\ \theta_y(t),\ \theta_z(t)$) is the sum of the i component responses:

$$\theta_x(t) = \sum_{i=0}^{\infty} \theta_{x\,i}(t) \qquad \theta_y(t) = \sum_{i=0}^{\infty} \theta_{y_i}(t) \qquad \theta_z(t) = \sum_{i=0}^{\infty} \theta_{z_i}(t) \tag{10.4.1-2}$$

$$\underline{\theta}(t) = \underline{u}_x\,\theta_x(t) + \underline{u}_y\,\theta_y(t) + \underline{u}_z\,\theta_z(t)$$

We seek the $\dot{\beta}_{m_z}$ response to the total angular vibration $\underline{\theta}(t)$ of (10.4.1-2). If Sections 10.1.1.1, 10.1.1.2 and 10.1.1.2.1 are reviewed, it will be recalled that $\dot{\beta}_{m_z}$ (the B Frame Z component of $\dot{\underline{\beta}}_m$) is derived from the integral of products between X and Y axis angular-displacement/angular-rate vibrations. Section 10.1.1.1 showed that if the frequencies for the X and Y axis angular vibrations were equal, a net coning attitude rate will develop around the Z axis. The $\dot{\beta}_{m_z}$ term represents the portion of the coning rate computed by the high speed Section 7.1.1.1.1 coning algorithm. Section 10.1.1.1 also showed that if the angular vibrations around the X and Y axes were of different frequencies, no net coning rate motion would develop. Based on these factors, consider what the Z axis coning response will be to $\theta_x(t)$ and $\theta_y(t)$ vibrations as defined in (10.4.1-2) and (10.4.1-1). The product of the X, Y effects will create a summation of products at the same and at different ω_i frequencies. The overall coning rate generated about the Z axis will only be produced from the products at the same frequency. Each of the same frequency products will generate an individual contribution to the net coning rate for its particular frequency ω_i. Then the total coning rate will be the sum of each of these i coning rate contributions. Thus, we can write:

$$\dot{\beta}_{m_z} = \sum_{i=0}^{\infty} \dot{\beta}_{m_{z/i}} \tag{10.4.1-3}$$

where

$\dot{\beta}_{m_{z/i}}$ = Contribution to $\dot{\beta}_{m_z}$ from the $\theta_{x\,i}(t)$, $\theta_{y_i}(t)$ vibration components.

A more rigorous derivation of (10.4.1-3) is too time consuming for presentation here. If desired, the reader can verify its authenticity as an analytical exercise.

An equation for $\dot{\beta}_{m_{z/i}}$ (based on the (10.4.1-1) angular vibration input) is readily obtained by inspection of the $\dot{\beta}_{m_z}$ result in Equations (10.3-20) (based on the (10.3-2) angular vibration input):

$$\dot{\beta}_{m_{z/i}} = \frac{1}{2}\,\omega_i\,B_{\theta_{x/i}}\,B_{\theta_{y/i}}\,p^2_{Vib_{o/i}}\,\sin\left(\phi_{\theta_{x/i}} - \phi_{\theta_{y/i}}\right)\left(1 - \frac{\sin\,\omega_i\,T_m}{\omega_i\,T_m}\right) \tag{10.4.1-4}$$

Then substituting (10.4.1-4) in (10.4.1-3) with rearrangement and application of (10.4-6) for $p_{Vib_{o/i}}$, we find for $\dot{\beta}_{m_z}$:

$$\dot{\beta}_{m_z} = \sum_{i=0}^{\infty} \omega_i \, B_{\theta_{x/i}} \, B_{\theta_{y/i}} \sin\left(\phi_{\theta_{x/i}} - \phi_{\theta_{y/i}}\right)\left(1 - \frac{\sin \omega_i T_m}{\omega_i T_m}\right) \frac{1}{\Delta\omega} \frac{1}{2}\left(a_{Vib_i}^2 + b_{Vib_i}^2\right) \Delta\omega \tag{10.4.1-5}$$

Equation (10.4.1-5) is now in a form in which we can account for random nature in $p_{Vib}(t)$ vibration through use of the expected value operator:

$$\mathcal{E}\left(\dot{\beta}_{m_z}\right) = \sum_{i=0}^{\infty} \omega_i \, B_{\theta_{x/i}} \, B_{\theta_{y/i}} \sin\left(\phi_{\theta_{x/i}} - \phi_{\theta_{y/i}}\right)\left(1 - \frac{\sin \omega_i T_m}{\omega_i T_m}\right) \frac{1}{\Delta\omega} \frac{1}{2} \mathcal{E}\left(a_{Vib_i}^2 + b_{Vib_i}^2\right) \Delta\omega \tag{10.4.1-6}$$

where

$\mathcal{E}(\,)$ = Expected value operator.

As an aside, we note that if the more general multi-axis $p_{Vib}(t)$ model had been used with independent components, the independence assumption would result in deleting all cross-axis product terms in Equation (10.4.1-6), leaving a summation for each $p_{Vib}(t)$ input axis component of identical form as the Equation (10.4.1-6) summation term. Hence, the effect of each independent $p_{Vib}(t)$ input axis component can be determined individually, as in this section, with the combined effect of all $p_{Vib}(t)$ input axis components obtained as the linear sum of the response for each.

As in Section 10.2.2, we now let T go to infinity in the limit which sets $\Delta\omega$ to the infinitesimal $d\omega$, and replace the i dependence in (10.4.1-6) by ω dependence. The result is:

$$\begin{aligned} \mathcal{E}\left(\dot{\beta}_{m_z}\right) = \int_0^{\infty} & \omega \, B_{\theta_x}(\omega) \, B_{\theta_y}(\omega) \sin\left(\phi_{\theta_x}(\omega) - \phi_{\theta_y}(\omega)\right)\Bigg(1 \\ & - \frac{\sin \omega T_m}{\omega T_m}\Bigg) \frac{1}{d\omega} \mathcal{E}\left[\frac{1}{2}\left(a_{Vib}(\omega)^2 + b_{Vib}(\omega)^2\right)\right] d\omega \end{aligned} \tag{10.4.1-7}$$

Finally, with Equation (10.2.2-22), we identify the $\frac{1}{d\omega} \mathcal{E}\left[\frac{1}{2}\left(a_{Vib}(\omega)^2 + b_{Vib}(\omega)^2\right)\right]$ term as the $p_{Vib}(t)$ random process power spectral density. Thus, (10.4.1-7) assumes the final form:

$$\mathcal{E}\left(\dot{\beta}_{m_z}\right) = \int_0^\infty \omega\, B_{\theta_x}(\omega)\, B_{\theta_y}(\omega)\, \sin\left(\phi_{\theta_x}(\omega) - \phi_{\theta_y}(\omega)\right)\left(1 - \frac{\sin\,\omega\,T_m}{\omega\,T_m}\right) G_{pV\,ib}(\omega)\, d\omega \tag{10.4.1-8}$$

where

$G_{pV\,ib}(\omega)$ = Power spectral density for the $p_{Vib}(t)$ random process.

In a typical INS technical specification, $G_{pV\,ib}(\omega)$ is a defined profile for test purposes. The $G_{pV\,ib}(\omega)$ profile presented in such specifications was probably derived from a form of Equation (10.2.2-21), with $\varphi_{pp}(\tau)$ computed from (10.2.2-9) using a representative ensemble of $p_{Vib}(t)$ samples.

For comparison with (10.4.1-8), let us now write the $\dot{\beta}_{m_z}$ discrete frequency result from Equations (10.3-20):

$$\dot{\beta}_{m_z} = \frac{1}{2}\,\Omega\, B_{\theta_x}\, B_{\theta_y}\, p_{Vib_0}^2\, \sin\left(\phi_{\theta_x} - \phi_{\theta_y}\right)\left(1 - \frac{\sin\,\Omega T_m}{\Omega T_m}\right) \tag{10.4.1-9}$$

Comparing (10.4.1-8) with (10.4.1-9) it should be obvious how to convert all of the summary Equation (10.3-20) discrete frequency performance parameters to their equivalent expected values under random vibration exposure; simply replace all Ω terms with ω, identify ω dependency for the amplitude ratios and phase angles, replace the $\frac{1}{2}\, p_{Vib_0}^2$ term with the power spectral density equivalent $G_{pV\,ib}(\omega)\, d\omega$, and integrate over the frequency spectrum from zero to infinity. Except for the $\delta \underline{R}_{SF/Algo}(t)$ response (treated separately in Section 10.4.2 to follow), the overall random response result for the Equation (10.3-20) parameters is:

$$\mathcal{E}\left(\dot{\Phi}_{Con_z}\right) = \int_0^\infty \omega\, B_{\theta_x}(\omega)\, B_{\theta_y}(\omega)\, \sin\left(\phi_{\theta_x}(\omega) - \phi_{\theta_y}(\omega)\right) G_{pVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\dot{\beta}_{m_z}\right) = \int_0^\infty \omega\, B_{\theta_x}(\omega)\, B_{\theta_y}(\omega)\, \sin\left(\phi_{\theta_x}(\omega) - \phi_{\theta_y}(\omega)\right)\left(1 - \frac{\sin\,\omega\, T_m}{\omega\, T_m}\right) G_{pVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\dot{\beta}_{Algo\text{-}m\,z}\right) = \int_0^\infty \omega\, B_{\theta_x}(\omega)\, B_{\theta_y}(\omega)\, \sin\left(\phi_{\theta_x}(\omega) - \phi_{\theta_y}(\omega)\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos\,\omega\, T_l\right)\right]\left(\frac{\sin\,\omega\, T_l}{\omega\, T_l}\right) - \frac{\sin\,\omega\, T_m}{\omega\, T_m}\right\} G_{pVib}(\omega)\, d\omega \tag{10.4.1-10}$$

$$\mathcal{E}\left(\delta\dot{\Phi}_{Algo\text{-}m\,z}\right) = \mathcal{E}\left(\delta\dot{\beta}_{Algo\text{-}m\,z}\right) = \int_0^\infty \omega\, B_{\theta_x}(\omega)\, B_{\theta_y}(\omega)\, \sin\left(\phi_{\theta_x}(\omega) - \phi_{\theta_y}(\omega)\right)\left\{\left[1 + \frac{1}{3}\left(1 - \cos\,\omega\, T_l\right)\right]\left(\frac{\sin\,\omega\, T_l}{\omega\, T_l}\right) - 1\right\} G_{pVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\dot{v}_{SF/Scul_z}\right) = \int_0^\infty \Big[B_{\theta_x}(\omega)\, B_{a_{SF_y}}(\omega)\, \cos\left(\phi_{\theta_x}(\omega) - \phi_{a_{SF_y}}(\omega)\right) - B_{\theta_y}(\omega)\, B_{a_{SF_x}}(\omega)\, \cos\left(\phi_{\theta_y}(\omega) - \phi_{a_{SF_x}}(\omega)\right)\Big] G_{pVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\Delta\dot{v}_{Scul\text{-}m_z}\right) = \int_0^\infty \Big[B_{\theta_x}(\omega)\, B_{a_{SF_y}}(\omega)\, \cos\left(\phi_{\theta_x}(\omega) - \phi_{a_{SF_y}}(\omega)\right) - B_{\theta_y}(\omega)\, B_{a_{SF_x}}(\omega)\, \cos\left(\phi_{\theta_y}(\omega) - \phi_{a_{SF_x}}(\omega)\right)\Big]\left(1 - \frac{\sin\,\omega\, T_m}{\omega\, T_m}\right) G_{pVib}(\omega)\, d\omega$$

(Continued)

$$\mathcal{E}\left(\Delta\dot{v}_{Scul/Algo\text{-}m_z}\right) = \int_0^\infty \Big[B_{\theta_x}(\omega)\, B_{a_{SF_y}}(\omega) \cos\left(\phi_{\theta_x}(\omega) - \phi_{a_{SF_y}}(\omega)\right)$$

$$- B_{\theta_y}(\omega)\, B_{a_{SF_x}}(\omega) \cos\left(\phi_{\theta_y}(\omega) - \phi_{a_{SF_x}}(\omega)\right)\Big] \left\{\left[1 + \frac{1}{3}\Big(1\right.\right.$$

$$\left.\left. - \cos\, \omega T_l\Big)\right]\left(\frac{\sin\, \omega T_l}{\omega T_l}\right) - \frac{\sin\, \omega T_m}{\omega T_m}\right\} G_{pVib}(\omega)\, d\omega \qquad \begin{array}{c}(10.4.1\text{-}10)\\ \text{(Continued)}\end{array}$$

$$\mathcal{E}\left(\delta\dot{v}_{SF/Algo\text{-}m_z}\right) = \mathcal{E}\left(\delta\Delta\dot{v}_{Scul/Algo\text{-}m_z}\right) = \int_0^\infty \Big[B_{\theta_x}(\omega)\, B_{a_{SF_y}}(\omega) \cos\left(\phi_{\theta_x}(\omega) - \phi_{a_{SF_y}}(\omega)\right)$$

$$- B_{\theta_y}(\omega)\, B_{a_{SF_x}}(\omega) \cos\left(\phi_{\theta_y}(\omega) - \phi_{a_{SF_x}}(\omega)\right)\Big] \left\{\left[1 + \frac{1}{3}\Big(1\right.\right.$$

$$\left.\left. - \cos\, \omega T_l\Big)\right]\left(\frac{\sin\, \omega T_l}{\omega T_l}\right) - 1\right\} G_{pVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\overline{f(t)}\right) = \int_0^\infty B_{h_x}(\omega)\, B_{h_y}(\omega) \cos\left(\phi_{h_x}(\omega) - \phi_{h_y}(\omega)\right) G_{pVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\overline{\delta a_{Accl_{G2}}}\right) = L_{G2} \int_0^\infty B_{Accl_{Inpt}}(\omega)\, B_{Accl_{Pend}}(\omega) \cos\Big(\phi_{aAccl_{Inpt}}(\omega)$$

$$- \phi_{aAccl_{Pend}}(\omega)\Big)\, G_{pVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\overline{\delta a_{Accl_{Aniso}}}\right) = L_{Aniso} \int_0^\infty B_{\omega_{IBAccl/Inpt}}(\omega)\, B_{\omega_{IBAccl/Pend}}(\omega) \cos\Big(\phi_{\omega_{IBAccl/Inpt}}(\omega)$$

$$- \phi_{\omega_{IBAccl/Pend}}(\omega)\Big)\, G_{pVib}(\omega)\, d\omega$$

(Continued)

$$\mathcal{E}\left(\delta \dot{v}_{SF/Scul/SnsDyn_z}\right) = \int_0^{\infty} \left\{ B_{\theta_x}(\omega)\, B_{aSF_y}(\omega) \left[B_{\omega ARS_x}(\omega)\, B_{aAccl_y}(\omega) \cos\left(\phi_{\theta_x}(\omega) \right.\right.\right.$$

$$\left. + \phi_{\omega ARS_x}(\omega) - \phi_{aSF_y}(\omega) - \phi_{aAccl_y}(\omega)\right) - \cos\left(\phi_{\theta_x}(\omega) - \phi_{aSF_y}(\omega)\right)\Big] \qquad \text{(10.4.1-10) (Continued)}$$

$$- B_{\theta_y}(\omega)\, B_{aSF_x}(\omega) \Big[B_{\omega ARS_y}(\omega)\, B_{aAccl_x}(\omega) \cos\left(\phi_{\theta_y}(\omega) + \phi_{\omega ARS_y}(\omega) - \phi_{aSF_x}(\omega)\right.$$

$$\left.\left. - \phi_{aAccl_x}(\omega)\right) - \cos\left(\phi_{\theta_y}(\omega) - \phi_{aSF_x}(\omega)\right)\Big]\right\} G_{pVib}(\omega)\, d\omega$$

We also note from Equation (10.2.2-24) that:

$$\mathcal{E}\left(\overline{p_{Vib}(t)^2}\right) = \int_0^{\infty} G_{pVib}(\omega)\, d\omega \qquad \text{(10.4.1-11)}$$

10.4.2 POSITION ALGORITHM ERROR RESPONSE TO RANDOM SYSTEM VIBRATION INPUT

To find the position algorithm error response to random vibration inputs, we begin with $\delta \underline{R}_{SF/Algo}(t)$ from Equations (10.3-20) and revise it to an equivalent format based on analytical equivalencies developed in Section 10.1.3.2.3. In particular, we first use Equations (10.1.3.2.3-15) and (10.1.3.2.3-18) to write:

$$\begin{aligned} &\sin\left(\Omega'(t - t_0) + \Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right) - \sin\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right) \\ &\quad = \cos\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right) \sin\left(\Omega'(t - t_0)\right) \\ &\qquad - \sin\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right)\left[1 - \cos\left(\Omega'(t - t_0)\right)\right] \qquad \text{(10.4.2-1)} \\ &\quad = \Omega'(t - t_0)\left[\cos\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right) f_1\left(\Omega'(t - t_0)\right)\right. \\ &\qquad \left. - \sin\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right) \Omega'(t - t_0)\, f_2\left(\Omega'(t - t_0)\right)\right] \end{aligned}$$

or the inverse:

$$\begin{aligned} &\cos\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right) f_1\left(\Omega'(t - t_0)\right) \\ &\qquad - \sin\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right) \Omega'(t - t_0)\, f_2\left(\Omega'(t - t_0)\right) \\ &\quad = \frac{1}{\Omega'(t - t_0)}\left[\sin\left(\Omega'(t - t_0) + \Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right)\right. \qquad \text{(10.4.2-2)} \\ &\qquad\qquad \left. - \sin\left(\Omega\, t_0 + \psi_{pVib} + \phi_{aSF}\right)\right] \end{aligned}$$

We also know from (10.1.3.2.3-16) that:

$$f_1(\Omega'\,T_l) \;=\; \frac{\sin\,\Omega'\,T_l}{\Omega'\,T_l} \tag{10.4.2-3}$$

Substituting (10.4.2-2) in the (10.3-20) $\delta\underline{R}_{SF/Algo}(t)$ expression, and (10.4.2-3) in the first $\frac{1}{12}$ expression in $\delta\underline{R}_{SF/Algo}(t)$, finds the alternate form:

$$\begin{aligned}\delta\underline{R}_{SF/Algo}(t) \;=\; -\;\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,p_{Vib_0}\,B_{aSF}\Bigg\{&\frac{\Omega}{\Omega'}\left(\frac{f_1(\Omega'\,T_l)}{2\,f_2(\Omega'\,T_l)} - \frac{\Omega'}{\Omega} + \frac{1}{12}\,\Omega'\,T_l\sin\,\Omega'\,T_l\right)\Big[\sin\big(\Omega'(t - t_0) + \Omega\,t_0 + \psi_{pVib} + \phi_{aSF}\big) \\ &- \sin\,(\Omega\,t_0 + \psi_{pVib} + \phi_{aSF})\Big] - \frac{1}{12}\,\Omega\,T_l\Big[\cos\big(\Omega'(t - t_0) + \Omega\,t_0 + \psi_{pVib} + \phi_{aSF}\big) \\ &- \cos\,(\Omega\,t_0 + \psi_{pVib} + \phi_{aSF})\Big]\,(1 - \cos\,\Omega'\,T_l)\Bigg\}\end{aligned} \tag{10.4.2-4}$$

$$\Omega' \;=\; \Omega - \frac{2\,\pi}{T_l}\left(\frac{\Omega\,T_l}{2\,\pi}\right)_{Intgr} \tag{10.4.2-5}$$

The reader should recognize the above process as a partial reversal of steps in Section 10.1.3.2.3 that led to the (10.1.3.2.3-19) singularity free $\delta\underline{R}_{SF/Algo}(t)$ form (from which the (10.3-20) $\delta\underline{R}_{SF/Algo}(t)$ version was derived). Let us now define parameters (some used previously) to simplify (10.4.2-4):

$$\begin{gathered}\phi \;\equiv\; \Omega\,t_0 + \psi_{pVib} + \phi_{aSF} \qquad\qquad \beta \;\equiv\; \Omega\,T_l \\ D \;\equiv\; \frac{\Omega}{\Omega'}\left(\frac{f_1(\Omega'\,T_l)}{2\,f_2(\Omega'\,T_l)} - \frac{\Omega'}{\Omega} + \frac{1}{12}\,\Omega'\,T_l\,\sin\,\Omega'\,T_l\right)\end{gathered} \tag{10.4.2-6}$$

With (10.4.2-6), Equation (10.4.2-4) is:

$$\begin{aligned}\delta\underline{R}_{SF/Algo}(t) \;=\; -\;\underline{u}_{Vib}\,\frac{1}{\Omega^2}\,p_{Vib_0}\,B_{aSF}\Big\{&D\left[\sin\big(\Omega'(t - t_0) + \phi\big) - \sin\,\phi\right] \\ &- \frac{1}{12}\,\beta\left[\cos\big(\Omega'(t - t_0) + \phi\big) - \cos\,\phi\right](1 - \cos\,\Omega'\,T_l)\Big\}\end{aligned} \tag{10.4.2-7}$$

We now equate $\delta\underline{R}_{SF/Algo}(t)$ to the sum of the responses to independent vibration inputs, each along the same vibration axis $\underline{u}_{Vib}$. Based on (10.4.2-5) - (10.4.2-7) we write:

$$\delta\underline{R}_{SF/Algo}(t) = \sum_i \delta\underline{R}_{SF/Algo_i}(t) \tag{10.4.2-8}$$

$$\begin{aligned}
\delta\underline{R}_{SF/Algo_i}(t) = &- \underline{u}_{Vib} \frac{1}{\omega_i^2} p_{Vib_{0/i}} B_{aSF_i} \Big\{ D_i \left[\sin\left(\omega_i'(t - t_0) + \phi_i\right) - \sin\phi_i \right] \\
&- \frac{1}{12} \beta_i \left[\cos\left(\omega_i'(t - t_0) + \phi_i\right) - \cos\phi_i \right] (1 - \cos\omega_i' T_l) \Big\}
\end{aligned}$$

$$\beta_i = \omega_i T_l \qquad \phi_i = \omega_i t_0 + \psi_{pVib_i} + \phi_{aSF_i} \qquad \omega_i' = \omega_i - \frac{2\pi}{T_l}\left(\frac{\omega_i T_l}{2\pi}\right)_{Intgr} \tag{10.4.2-9}$$

$$D_i = \frac{\omega_i}{\omega_i'}\left(\frac{f_1(\omega_i' T_l)}{2 f_2(\omega_i' T_l)} - \frac{\omega_i'}{\omega_i} + \frac{1}{12}\omega_i' T_l \sin\omega_i' T_l\right)$$

where

$\delta\underline{R}_{SF/Algo_i}(t)$ = The response of $\delta\underline{R}_{SF/Algo}(t)$ to the $p_{Vib_i}(t)$ vibration of Equation (10.4-1) and (10.4-5) at frequency ω_i.

ω_i' = Folded version of ω_i.

The response of $\delta\underline{R}_{SF/Algo}(t)$ to random inputs will be calculated in the form of the variance about the mean of its "signed magnitude" which, with (10.4.2-8), is defined by:

$$\delta R_{SF/Algo}(t) \equiv \underline{u}_{Vib} \cdot \delta\underline{R}_{SF/Algo}(t) \qquad \delta R_{SF/Algo_i}(t) \equiv \underline{u}_{Vib} \cdot \delta\underline{R}_{SF/Algo_i}(t)$$

$$\delta R_{SF/Algo}(t) = \sum_i \left(\underline{u}_{Vib} \cdot \delta\underline{R}_{SF/Algo_i}(t)\right) = \sum_i \delta R_{SF/Algo_i}(t) \tag{10.4.2-10}$$

where

$\delta R_{SF/Algo}(t)$, $\delta R_{SF/Algo_i}(t)$ = The "signed magnitudes" of $\delta\underline{R}_{SF/Algo}(t)$ and $\delta\underline{R}_{SF/Algo_i}(t)$ defined as their projections along the vibration axis.

Using (10.4.2-8), and assuming that $\delta R_{SF/Algo}(t)$ has zero mean value (i.e., purely oscillatory in (10.4.2-8) - (10.4.2-9) with random $p_{Vib_i}(t)$ phase angle ψ_{pVib_i}), the variance of $\delta R_{SF/Algo}(t)$ about its mean is from (10.4.2-10):

$$\begin{aligned} \mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right) &= \mathcal{E}\left[\left(\sum_i \delta R_{SF/Algo_i}(t)\right)\left(\sum_j \delta R_{SF/Algo_j}(t)\right)\right] \\ &= \sum_i \mathcal{E}\left(\delta R^2_{SF/Algo_i}(t)\right) + 2\sum_{i \neq j}\sum_j \mathcal{E}\left(\delta R_{SF/Algo_i}(t)\, \delta R_{SF/Algo_j}(t)\right) \end{aligned} \tag{10.4.2-11}$$

where

$\mathcal{E}(\,)$ = Expected value operator defined as an average at the particular time t across the ensemble of possible $p_{Vib_i}(t)$ vibration components (i.e., each i component having a random possible amplitude and phase angle).

j = Dummy index for i.

We assume that the i vibration components are statistically independent from one another so that the expected value of products between unequal i components is zero. Then (10.4.2-11) simplifies to:

$$\mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right) = \sum_i \mathcal{E}\left(\delta R^2_{SF/Algo_i}(t)\right) \tag{10.4.2-12}$$

At this point we make some assumptions regarding the statistical characteristics of the (10.4-1) and (10.4-5) $p_{Vib_i}(t)$ vibration components. In particular, we define the $p_{Vib_{0/i}}$ amplitude and ψ_{pVib_i} phase to be statistically independent of one another so that we can write:

$$P(p_{Vib_{0/i}}, \psi_{pVib_i}) = P(p_{Vib_{0/i}})\, P(\psi_{pVib_i}) \tag{10.4.2-13}$$

where

$P(p_{Vib_{0/i}}, \psi_{pVib_i})$ = Joint probability density function associated with the random parameters $p_{Vib_{0/i}}$ and ψ_{pVib_i}. The probability that $p_{Vib_{0/i}}$ and ψ_{pVib_i} will lie in the infinitesimally small interval $dp_{Vib_{0/i}}$ and $d\psi_{pVib_i}$ over the range of possible $p_{Vib_{0/i}}$ and ψ_{pVib_i} values, equals $P(p_{Vib_{0/i}}, \psi_{pVib_i})\, dp_{Vib_{0/i}}\, d\psi_{pVib_i}$.

$P(p_{Vib_{0/i}}), P(\psi_{pVib_i})$ = Individual probability density functions associated with $p_{Vib_{0/i}}$ and ψ_{pVib_i}. $P(p_{Vib_{0/i}})\ dp_{Vib_{0/i}}$ is the probability that $p_{Vib_{0/i}}$ lies in the infinitesimally small interval $dp_{Vib_{0/i}}$; $P(\psi_{pVib_i})\ d\psi_{pVib_i}$ is the probability that ψ_{pVib_i} lies in the infinitesimally small interval $d\psi_{pVib_i}$.

Based on the definition for $P(p_{Vib_{0/i}},\psi_{pVib_i})$, the expected value for an arbitrary function to lie in the $dp_{Vib_{0/i}}$ and $d\psi_{pVib_i}$ intervals equals the function multiplied by $P(p_{Vib_{0/i}},\psi_{pVib_i})\ d\psi_{pVib_i}\ dp_{Vib_{0/i}}$. Then the expected value of the function over the range of possible $p_{Vib_{0/i}}$ and ψ_{pVib_i} values is the double integral of the function multiplied by $P(p_{Vib_{0/i}},\psi_{pVib_i})$ over these ranges. We define the range of $p_{Vib_{0/i}}$ to be from zero to plus infinity and the range of ψ_{pVib_i} to be from $-\pi$ to $+\pi$. For our case, the particular function we will be analyzing will be a function of $p_{Vib_{0/i}}$ multiplied by a function of ψ_{pVib_i}. Thus, using (10.4.2-13) we can write:

$$\mathcal{E}\left(f(p_{Vib_{0/i}})\ g(\psi_{pVib_i})\right)$$

$$= \int_0^{\infty}\int_{-\pi}^{\pi} f(p_{Vib_{0/i}})\ g(\psi_{pVib_i})\ P(p_{Vib_{0/i}})\ P(\psi_{pVib_i})\ d\psi_{pVib_i}\ dp_{Vib_{0/i}} \tag{10.4.2-14}$$

$$= \int_0^{\infty} f(p_{Vib_{0/i}})\ P(p_{Vib_{0/i}})\ dp_{Vib_{0/i}} \int_{-\pi}^{\pi} g(\psi_{pVib_i})\ P(\psi_{pVib_i})\ d\psi_{pVib_i}$$

where

f( ), g( ) = General functions of the variable in brackets.

But from the definitions for $P(p_{Vib_{0/i}})$ and $P(\psi_{pVib_i})$, the expected values for $f(p_{Vib_{0/i}})$ and $g(\psi_{pVib_i})$ are the single integral terms in (10.4.2-14). Thus, (10.4.2-14) is:

$$\mathcal{E}\left(f(p_{Vib_{0/i}})\ g(\psi_{pVib_i})\right) = \mathcal{E}\left(f(p_{Vib_{0/i}})\right)\mathcal{E}\left(g(\psi_{pVib_i})\right) \tag{10.4.2-15}$$

with

$$\mathcal{E}\left(f(p_{Vib_{0/i}})\right) = \int_0^{\infty} f(p_{Vib_{0/i}})\ P(p_{Vib_{0/i}})\ dp_{Vib_{0/i}} \tag{10.4.2-16}$$

$$\mathcal{E}\left(g(\psi_{pVib_i})\right) = \int_{-\pi}^{\pi} g(\psi_{pVib_i})\, P(\psi_{pVib_i})\, d\psi_{pVib_i} \tag{10.4.2-17}$$

Using ϕ_i as defined in (10.4.2-9), we can also write:

$$g(\psi_{pVib_i}) = h(\phi_i) \tag{10.4.2-18}$$

where

$h(\phi_i)$ = General function of ϕ_i that is independent of $p_{Vib_{0/i}}$.

Based on (10.4.2-18), (10.4.2-15) is equivalently:

$$\mathcal{E}\left(f(p_{Vib_{0/i}})\, h(\phi_i)\right) = \mathcal{E}\left(f(p_{Vib_{0/i}})\right) \mathcal{E}\left(h(\phi_i)\right) \tag{10.4.2-19}$$

We can also write for $P(\psi_{pVib_i})$ over the range of ψ_{pVib_i}:

$$\int_{-\pi}^{\pi} P(\psi_{pVib_i})\, d\psi_{pVib_i} = 1 \tag{10.4.2-20}$$

which states that the probability is one for ψ_{pVib_i} to be in its defined range. Assuming that ψ_{pVib_i} has equal probability (i.e., constant probability density) of being anywhere in the $-\pi$ to $+\pi$ interval, (10.4.2-20) shows that:

$$P(\psi_{pVib_i}) = \frac{1}{2\pi} \tag{10.4.2-21}$$

Using (10.4.2-21) for $P(\psi_{pVib_i})$, (10.4.2-17) becomes:

$$\mathcal{E}\left(g(\psi_{pVib_i})\right) = \frac{1}{2\pi} \int_{-\pi}^{\pi} g(\psi_{pVib_i})\, d\psi_{pVib_i} \tag{10.4.2-22}$$

Using (10.4.2-22), we see that:

$$\begin{gathered}
\mathcal{E}\left(\sin(A + \psi_{pVib_i})\right) = 0 \qquad \mathcal{E}\left(\cos(A + \psi_{pVib_i})\right) = 0 \\
\mathcal{E}\left(\sin(A + \psi_{pVib_i})\cos(A + \psi_{pVib_i})\right) = 0 \\
\mathcal{E}\left(\sin^2(A + \psi_{pVib_i})\right) = \frac{1}{2} \qquad \mathcal{E}\left(\cos^2(A + \psi_{pVib_i})\right) = \frac{1}{2}
\end{gathered} \tag{10.4.2-23}$$

where

$$A = \text{General parameter that is independent of } \psi_{pVib_i}.$$

Equations (10.4.2-23) can be used to find the expected value of more complicated expressions that we will need in developing our position error response to random vibrations. For example, consider the function:

$$\begin{aligned}
&\left(\sin(B + \phi_i) - \sin\phi_i\right)\left(\cos(B + \phi_i) - \cos\phi_i\right) \\
&\quad = \sin(B + \phi_i)\cos(B + \phi_i) - \sin(B + \phi_i)\cos\phi_i \\
&\qquad\quad - \sin\phi_i\cos(B + \phi_i) + \sin\phi_i\cos\phi_i \\
&\quad = \sin(B + \phi_i)\cos(B + \phi_i) - \sin B\cos^2\phi_i - \cos B\sin\phi_i\cos\phi_i \\
&\qquad\quad - \cos B\sin\phi_i\cos\phi_i + \sin B\sin^2\phi_i + \sin\phi_i\cos\phi_i \\
&\quad = \sin(B + \phi_i)\cos(B + \phi_i) + \sin B\left(\sin^2\phi_i - \cos^2\phi_i\right) + (1 - 2\cos B)\sin\phi_i\cos\phi_i
\end{aligned} \tag{10.4.2-24}$$

in which ϕ_i is as defined in (10.4.2-9) and where

$$B = \text{General parameter that is independent of } \phi_i.$$

Substituting ϕ_i from (10.4.2-9) in (10.4.2-24) finds:

$$\begin{aligned}
&\left(\sin(B + \phi_i) - \sin\phi_i\right)\left(\cos(B + \phi_i) - \cos\phi_i\right) \\
&\quad = \sin\left(A + B + \psi_{pVib_i}\right)\cos\left(A + B + \psi_{pVib_i}\right) \\
&\qquad + \sin B\left(\sin^2(A + \psi_{pVib_i}) - \cos^2(A + \psi_{pVib_i})\right) \\
&\qquad + (1 - 2\cos B)\sin(A + \psi_{pVib_i})\cos(A + \psi_{pVib_i})
\end{aligned} \tag{10.4.2-25}$$

in which

$$A = \omega_i\, t_0 + \phi_{aSF_i} \tag{10.4.2-26}$$

Taking the expected value of (10.4.2-25) then shows with (10.4.2-23) that:

$$\mathcal{E}\left[\left(\sin(B + \phi_i) - \sin\phi_i\right)\left(\cos(B + \phi_i) - \cos\phi_i\right)\right] = 0 \tag{10.4.2-27}$$

Consider the function $\left(\sin(B + \phi_i) - \sin\phi_i\right)^2$. First we note that:

$$\begin{aligned}
\sin(B + \phi_i) - \sin\phi_i &= \sin B\cos\phi_i + \cos B\sin\phi_i - \sin\phi_i \\
&= \sin B\cos\phi_i - \sin\phi_i\,(1 - \cos B)
\end{aligned} \tag{10.4.2-28}$$

so that

$$\left(\sin (B + \phi_i) - \sin \phi_i\right)^2 = \sin^2 B \cos^2 \phi_i - 2 \sin \phi_i \cos \phi_i (1 - \cos B) \sin B + \sin^2 \phi_i (1 - \cos B)^2 \tag{10.4.2-29}$$

Substituting ϕ_i from (10.4.2-9) in (10.4.2-29), taking the expected value, and applying (10.4.2-23) then obtains:

$$\begin{aligned} \mathcal{E}\left[\left(\sin (B + \phi_i) - \sin \phi_i\right)^2\right] &= \frac{1}{2} \sin^2 B + \frac{1}{2} (1 - \cos B)^2 \\ &= \frac{1}{2}\left(\sin^2 B + 1 - 2 \cos B + \cos^2 B\right) = 1 - \cos B \end{aligned} \tag{10.4.2-30}$$

In a similar manner we find:

$$\mathcal{E}\left[\left(\cos (B + \phi_i) - \cos \phi_i\right)^2\right] = 1 - \cos B \tag{10.4.2-31}$$

We are now ready to apply our results in determining the expected value of $\delta R^2_{SF/Alg_i}(t)$. First we write from (10.4.2-9) and (10.4.2-10):

$$\begin{aligned} \delta R_{SF/Algo_i}(t) &= - F_i H_i \\ F_i &\equiv \frac{1}{\omega_i^2} p_{Vib_{0/i}} B_{aSF_i} \\ H_i &\equiv D_i \left[\sin\left(\omega_i'(t - t_0) + \phi_i\right) - \sin \phi_i\right] \\ &\quad - \frac{1}{12} \beta_i \left[\cos\left(\omega_i'(t - t_0) + \phi_i\right) - \cos \phi_i\right] (1 - \cos \omega_i' T_l)^2 \end{aligned} \tag{10.4.2-32}$$

from which we get:

$$\begin{aligned} \delta R^2_{SF/Algo_i}(t) &= f(p_{Vib_{0/i}})\, h(\phi_i) \\ f(p_{Vib_{0/i}}) &\equiv F_i^2 = \frac{1}{\omega_i^4} p^2_{Vib_{0/i}} B^2_{aSF_i} \\ h(\phi_i) &\equiv H_i^2 = D_i^2 \left[\sin\left(\omega_i'(t - t_0) + \phi_i\right) - \sin \phi_i\right]^2 \\ &\quad - \frac{1}{6} D_i \beta_i \left[\sin\left(\omega_i'(t - t_0) + \phi_i\right) - \sin \phi_i\right]\left[\cos\left(\omega_i'(t - t_0) + \phi_i\right) - \cos \phi_i\right] (1 - \cos \omega_i' T_l) \\ &\quad + \frac{1}{144} \beta_i^2 \left[\cos\left(\omega_i'(t - t_0) + \phi_i\right) - \cos \phi_i\right]^2 (1 - \cos \omega_i' T_l)^2 \end{aligned} \tag{10.4.2-33}$$

The expected value of $\delta R^2_{SF/Algo_i}(t)$ in (10.4.2-12) is then found by applying (10.4.2-19) to (10.4.2-33) with (10.4.2-27), (10.4.2-30) and (10.4.2-31):

$$\mathcal{E}\left(\delta R^2_{SF/Algo_i}(t)\right) = \frac{1}{\omega_i^4}\,\mathcal{E}\,(p^2_{Vib_{0/i}})\,B^2_{aSF_i}\Big(D_i^2 + \frac{1}{144}\,\beta_i^2\,(1 - \cos\,\omega_i'\,T_l)^2\Big)\Big[1 - \cos\Big(\omega_i'(t - t_0)\Big)\Big] \tag{10.4.2-34}$$

Returning to (10.4.2-9) for the D_i definition, we write:

$$D_i = \frac{\omega_i}{\omega_i'}\left(E_i + \frac{1}{12}\,\omega_i'\,T_l \sin\,\omega_i'\,T_l\right) \qquad E_i \equiv \frac{f_1(\omega_i'\,T_l)}{2\,f_2(\omega_i'\,T_l)} - \frac{\omega_i'}{\omega_i} \tag{10.4.2-35}$$

Using (10.4.2-35) and the (10.4.2-9) β_i definition, the middle term in (10.4.2-34) becomes:

$$\begin{aligned}
D_i^2 &+ \frac{1}{144}\,\beta_i^2\,(1 - \cos\,\omega_i'\,T_l)^2 \\
&= \frac{\omega_i^2}{\omega_i'^2}\,E_i^{\,2} + 2\,\frac{\omega_i^2}{\omega_i'^2}\,E_i\left(\frac{1}{12}\,\omega_i'\,T_l \sin\,\omega_i'\,T_l\right) \\
&\qquad + \frac{\omega_i^2}{\omega_i'^2}\,\frac{1}{144}\,\omega_i'^2\,T_l^2\,\sin^2\omega_i'\,T_l + \frac{1}{144}\,\beta_i^2\,(1 - \cos\,\omega_i'\,T_l)^2 \\
&= \frac{\omega_i^2}{\omega_i'^2}\,E_i^{\,2} + 2\,E_i\,\frac{\omega_i}{\omega_i'}\left(\frac{1}{12}\,\beta_i \sin\,\omega_i'\,T_l\right) \\
&\qquad + \frac{1}{144}\,\beta_i^2\,\sin^2\omega_i'\,T_l + \frac{1}{144}\,\beta_i^2\,(1 - \cos\,\omega_i'\,T_l)^2 \\
&= \frac{\omega_i^2}{\omega_i'^2}\,E_i^{\,2} + \frac{1}{6}\,E_i\,\frac{\omega_i}{\omega_i'}\,\beta_i \sin\,\omega_i'\,T_l \\
&\qquad + \frac{1}{144}\,\beta_i^2\left(\sin^2\omega_i'\,T_l + (1 - \cos\,\omega_i'\,T_l)^2\right)
\end{aligned} \tag{10.4.2-36}$$

The last bracketed term in (10.4.2-36) simplifies to:

$$\begin{aligned}
\sin^2\omega_i'\,T_l + (1 - \cos\,\omega_i'\,T_l)^2 &= \sin^2\omega_i'\,T_l + 1 - 2\cos\,\omega_i'\,T_l + \cos^2\omega_i'\,T_l \\
&= 2\,(1 - \cos\,\omega_i'\,T_l)
\end{aligned} \tag{10.4.2-37}$$

Using (10.4.2-37), Equation (10.4.2-36) reduces:

$$
\begin{aligned}
&D_i^2 + \frac{1}{144}\,\beta_i^2\,(\,1 - \cos\,\omega_i'\,T_l)^2 \\
&\quad = \frac{\omega_i^2}{\omega_i'^2}\,E_i^{\,2} + \frac{1}{6}\,E_i\,\frac{\omega_i}{\omega_i'}\,\beta_i\,\sin\,\omega_i'\,T_l + \frac{1}{72}\,\beta_i^2\,(1 - \cos\,\omega_i'\,T_l)
\end{aligned}
\tag{10.4.2-38}
$$

With (10.4.2-38) and (10.4.2-9) for β_i, (10.4.2-34) becomes:

$$
\begin{aligned}
\mathcal{E}\left(\delta R^2_{SF/Algo_i}(t)\right) &= \frac{1}{\omega_i^4}\,\mathcal{E}(p^2_{Vib_{0/i}})\,B^2_{aSF_i}\Bigg(\frac{\omega_i^2}{\omega_i'^2}\,E_i^{\,2} \\
&+ \frac{1}{6}\,E_i\,\frac{\omega_i^2}{\omega_i'}\,T_l\sin\,\omega_i'\,T_l + \frac{1}{72}\,(\omega_i\,T_l)^2\,(1 - \cos\,\omega_i'\,T_l)\Bigg)\left[1 - \cos\left(\omega_i'(t - t_0)\right)\right]
\end{aligned}
\tag{10.4.2-39}
$$

Equation (10.4.2-39) is now in a convenient form for introducing the (10.1.3.2.3-16) functions to eliminate ω_i' singularities. As in (10.1.3.2.3-17) and (10.1.3.2.3-18) we write:

$$
\begin{aligned}
&\sin\,\omega_i'\,T_l \;=\; \omega_i'\,T_l\,f_1(\omega_i'\,T_l) \qquad 1 - \cos\,\omega_i'\,T_l \;=\; (\omega_i'\,T_l)^2\,f_2(\omega_i'\,T_l) \\
&\qquad 1 - \cos\left(\omega_i'(t - t_0)\right) \;=\; \left(\omega_i'\,(t - t_0)\right)^2\,f_2\left(\omega_i'(t - t_0)\right)
\end{aligned}
\tag{10.4.2-40}
$$

Applying (10.4.2-40) in (10.4.2-39) with rearrangement yields the desired singularity free form for $\mathcal{E}\left(\delta R^2_{SF/Algo_i}(t)\right)$:

$$
\begin{aligned}
\mathcal{E}\left(\delta R^2_{SF/Algo_i}(t)\right) &= (t - t_0)^2\,B^2_{aSF_i}\,\frac{1}{\omega_i^2}\Bigg\{E_i^{\,2} + \frac{1}{6}\,(\omega_i'\,T_l)^2\Big[E_i\,f_1(\omega_i'\,T_l) \\
&+ \frac{1}{12}\,(\omega_i'\,T_l)^2\,f_2(\omega_i'\,T_l)\Big]\Bigg\}\,f_2\left(\omega_i'(t - t_0)\right)\,\mathcal{E}(p^2_{Vib_{0/i}})
\end{aligned}
\tag{10.4.2-41}
$$

We substitute (10.4.2-41) in (10.4.2-12) with (10.4-6) for $p_{Vib_{0/i}}$ to obtain for the $\delta R_{SF/Algo}(t)$ variance:

$$
\begin{aligned}
\mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right) &= (t - t_0)^2\sum_i B^2_{aSF_i}\,\frac{1}{\omega_i^2}\Bigg\{E_i^{\,2} + \frac{1}{6}\,(\omega_i'\,T_l)^2\Big[E_i\,f_1(\omega_i'\,T_l) \\
&+ \frac{1}{12}\,(\omega_i'\,T_l)^2\,f_2(\omega_i'\,T_l)\Big]\Bigg\}\,f_2\left(\omega_i'(t - t_0)\right)\,\frac{1}{\Delta\omega}\,\mathcal{E}\left(a^2_{Vib_i} + b^2_{Vib_i}\right)\Delta\omega
\end{aligned}
\tag{10.4.2-42}
$$

in which $\Delta\omega$ as has been defined previously in Section 10.4 by Equations (10.4-2). In the limit as $\Delta\omega \rightarrow d\omega$, (10.4.2-42) becomes:

$$\mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right) = (t - t_0)^2 \int_0^\infty B^2_{aSF}(\omega)\, \frac{2}{\omega^2} \left\{ E(\omega)^2 + \frac{1}{6}\,(\omega'\, T_l)^2 \left[E(\omega)\, f_1(\omega'\, T_l) \right.\right.$$

$$\left.\left. + \frac{1}{12}\,(\omega'\, T_l)^2\, f_2(\omega'\, T_l) \right] \right\} f_2\big(\omega'(t - t_0)\big)\, \frac{1}{d\omega}\, \mathcal{E}\left[\frac{1}{2}\left(a^2_{Vib}(\omega) + b^2_{Vib}(\omega)\right)\right] d\omega \qquad (10.4.2\text{-}43)$$

Finally, with Equation (10.2.2-22), we identify the $\frac{1}{d\omega}\, \mathcal{E}\left[\frac{1}{2}\left(a^2_{Vib}(\omega) + b^2_{Vib}(\omega)\right)\right]$ term as the $p_{Vib}(t)$ random process power spectral density so that with (10.4.2-35) and (10.4.2-9) in the limit for $E(\omega)$ and ω':

$$\mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right) = (t - t_0)^2 \int_0^\infty B^2_{aSF}(\omega)\, \frac{2}{\omega^2} \left\{ E(\omega)^2 + \frac{1}{6}\,(\omega'\, T_l)^2 \left[E(\omega)\, f_1(\omega'\, T_l) \right.\right.$$

$$\left.\left. + \frac{1}{12}\,(\omega'\, T_l)^2\, f_2(\omega'\, T_l) \right] \right\} f_2\big(\omega'(t - t_0)\big)\, G_{pVib}(\omega)\, d\omega \qquad (10.4.2\text{-}44)$$

$$\omega' = \omega - \frac{2\,\pi}{T_l}\left(\frac{\omega\, T_l}{2\,\pi}\right)_{Intgr} \qquad\qquad E(\omega) = \frac{f_1(\omega'\, T_l)}{2\, f_2(\omega'\, T_l)} - \frac{\omega'}{\omega}$$

The f_1 and f_2 functions in (10.4.2-44) are defined by Equations (10.1.3.2.3-16).

Equations (10.4.2-44) provide a singularity free version of the $\mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right)$ response to the $G_{pVib}(\omega)$ random input vibration power spectrum for addition to the Equations (10.4.1-10) overall random response performance summary list.

10.5 SYSTEM DYNAMIC RESPONSE ANALYSIS MODEL

The results of Sections 10.3 and 10.4 summarized in Equations (10.3-20), (10.4.1-10) and (10.4.2-44) are based on having knowledge of the INS sensor assembly dynamic response to applied $p_{Vib}(t)$ input vibrations as represented by the $B(\omega)$, $\phi(\omega)$ amplitude-ratio/phase-angle terms. Finding values for these terms can be a time consuming computer aided software design process involving complex mechanical modeling of the INS structure and how it mechanically couples to the $p_{Vib}(t)$ vibration source. Due to its complexity, the process is inherently prone to data input error that distorts results obtained. To provide a reasonableness check on the results, simplified dynamic models are frequently employed for comparison that lend themselves to closed-form analytical solutions. Once the detailed modeling results match the simplified model within its approximation uncertainty, the detailed model is deemed valid for use in estimating $B(\omega)$, $\phi(\omega)$.

From a broader perspective, it must be recognized that it is virtually impossible to develop an accurate mechanical dynamic model for an INS in a user vehicle due to variations in mechanical structural properties between INS's of a particular design (e.g., variations in stiffness/damping characteristics of electronic circuit boards in their respective card guides, variations in mechanical housings, variations in mounting interfaces, etc.), as well as variations in the characteristics for a particular INS over temperature and time. On the other hand, for performance analysis purposes, only "ball-park" accuracy is generally required for the $B(\omega)$, $\phi(\omega)$ dynamic characteristics. All things considered, it becomes reasonable to use the simplified analytical models for $B(\omega)$, $\phi(\omega)$, thereby eliminating the need for cumbersome computerized modeling.

In this section we will develop simplified analytical models depicting the INS sensor response to vibration excitation. These models will then serve as the basis for the simulation analysis tool described in Section 10.6 for estimating numerical values of the Section 10.3 and 10.4 vibration performance parameters under prescribed system input vibration exposures. Two simplified models will be considered; one based on the INS sensor assembly dynamic response to a linear system input forcing function; the other based on the response to a rotary input forcing function.

10.5.1 DYNAMIC MODEL RESPONSE TO LINEAR SYSTEM FORCING FUNCTION

The simplified dynamic model with linear system input forcing is based on the Figure 10.5.1-1 sketch illustrating the response of the INS sensor assembly to prescribed linear input motion of the INS mount.

In Figure 10.5.1-1, the sensor assembly (round shaped object) is shown interfaced to the INS mount (the vertical "wall") through two elastomeric isolators with identified spring/damping characteristics (k_1, k_2 and c_1, c_2). The sensor assembly has freedom to translate to the left and right, and to rotate in the plane of the page. The mount interface "wall" has a prescribed dynamic position displacement that drives the sensor assembly motion through the isolators. Nominally, the isolators have equal spring damping characteristics, and are attached to the sensor assembly at equal distances from the center of mass (i.e., what is known as a CG

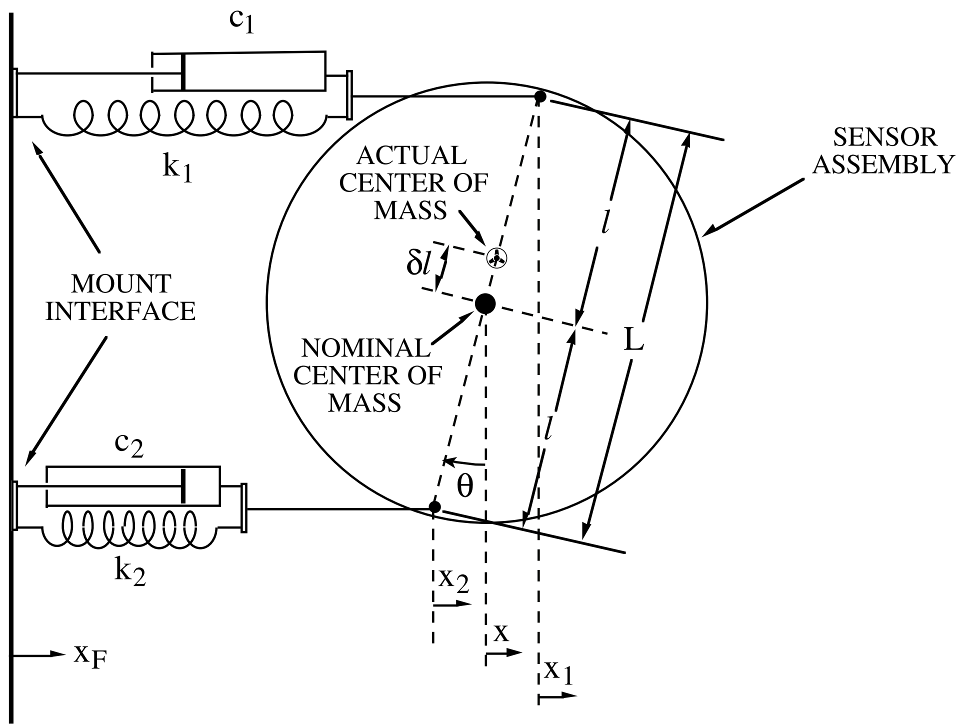


Figure 10.5.1-1 Sensor Assembly Dynamic Response to Linear Forcing

mount). For such an arrangement, translational motion of the mount wall generates no net torque around the sensor assembly center of mass. Nominally, then, no rotation will be generated under wall mount linear vibration. Sensor assembly rotation will be generated from asymmetries in the isolators (variations in the spring/damping characteristics) and variations in the lever arm distances between the isolators and the sensor assembly center of mass. Based on the Figure 10.5.1-1 model and assuming small angle response, we write the classical Newtonian equations relating the sensor assembly angular acceleration to applied torques around the center of mass, and the sensor assembly center of mass acceleration to the applied forces:

$$m \frac{d^2(x + \theta\, \delta l)}{dt^2} = - c_1 \left(\dot{x}_1 - \dot{x}_F\right) - k_1 \left(x_1 - x_F\right) - c_2 \left(\dot{x}_2 - \dot{x}_F\right) - k_2 \left(x_2 - x_F\right) \tag{10.5.1-1}$$

$$J\, \ddot{\theta} = - \left[c_1 \left(\dot{x}_1 - \dot{x}_F\right) + k_1 \left(x_1 - x_F\right)\right] \left(l - \delta l\right) + \left[c_2 \left(\dot{x}_2 - \dot{x}_F\right) + k_2 \left(x_2 - x_F\right)\right] \left(l + \delta l\right) \tag{10.5.1-2}$$

with

$$x_1 = x + l\,\theta \qquad x_2 = x - l\,\theta \tag{10.5.1-3}$$

where

x_F = Linear distance movement of the sensor mount wall from its neutral position under forced "F" input motion of the sensor mount interface wall. The neutral position is defined as the position of the sensor mount interface wall under zero dynamic wall motion.

θ = Sensor assembly angular response to x_F from its neutral angular orientation. The neutral angular orientation is defined as the orientation of the sensor assembly under zero dynamic motion of the sensor mount interface wall.

x = Linear distance movement response to x_F of the sensor assembly nominal center of mass point from its neutral position. The neutral position is defined as the position of the nominal center of mass point under zero dynamic motion of the sensor mount interface wall.

x_1, x_2 = Linear distance movement response to x_F of the sensor assembly isolator attachment points from their neutral position. The sensor assembly isolator attachment point neutral positions are defined as the position of the attachment points under zero dynamic motion of the sensor mount interface wall.

k_1, c_1, k_2, c_2 = Spring, damping coefficients for the upper and lower isolators.

l = Nominal distance from each sensor-assembly isolator attachment point to the nominal sensor assembly center of mass.

δl = Distance between the nominal and actual sensor assembly centers of mass.

J = Moment of inertia of the sensor assembly about its actual center of mass.

m = Mass of the sensor assembly.

Equations (10.5.1-1) - (10.5.1-2) assume that the axis of rotation is a principal moment of inertia axis so that product of inertia terms do not appear (Reference 8 - Section 5-3). The $\theta\,\delta l$ term in (10.5.1-1) is the small displacement of the actual relative to the nominal center of mass (based on θ being small), thus $x + \theta\,\delta l$ is the actual center of mass total displacement. We intuitively expect the $\theta\,\delta l$ term to be second order, hence, negligible, but will carry it for a while to check our intuition. Now, let's define:

$$\begin{aligned} k &\equiv \frac{1}{2}\left(k_1 + k_2\right) \qquad & c &\equiv \frac{1}{2}\left(c_1 + c_2\right) \\ \delta k &\equiv k_2 - k_1 \qquad & \delta c &\equiv c_2 - c_1 \end{aligned} \tag{10.5.1-4}$$

where

k, c = Average isolator spring, damping coefficients.

$\delta k, \delta c$ = Spring/damping mismatch between the isolators.

From (10.5.1-4) we see that:

$$k_1 = k - \frac{1}{2}\delta k \qquad k_2 = k + \frac{1}{2}\delta k$$
$$c_1 = c - \frac{1}{2}\delta c \qquad c_2 = c + \frac{1}{2}\delta c \tag{10.5.1-5}$$

Substituting (10.5.1-3) and (10.5.1-5) into (10.5.1-1) and grouping terms gives:

$$\begin{aligned} m\frac{d^2(x + \theta\,\delta l)}{dt^2} &= m\ddot{x} + m\,\delta l\,\ddot{\theta} \\ &= -\left(c - \frac{1}{2}\delta c\right)\left(\dot{x} + l\,\dot{\theta} - \dot{x}_F\right) - \left(k - \frac{1}{2}\delta k\right)\left(x + l\,\theta - x_F\right) \\ &\quad -\left(c + \frac{1}{2}\delta c\right)\left(\dot{x} - l\,\dot{\theta} - \dot{x}_F\right) - \left(k + \frac{1}{2}\delta k\right)\left(x - l\,\theta - x_F\right) \\ &= -2c\left(\dot{x} - \dot{x}_F\right) - 2k\left(x - x_F\right) + \delta c\, l\,\dot{\theta} + \delta k\, l\,\theta \end{aligned} \tag{10.5.1-6}$$

or upon rearrangement:

$$m\ddot{x} + 2c\dot{x} + 2kx = 2c\dot{x}_F + 2kx_F - m\,\delta l\,\ddot{\theta} + \delta c\, l\,\dot{\theta} + \delta k\, l\,\theta \tag{10.5.1-7}$$

Applying (10.5.1-3) and (10.5.1-5) to (10.5.1-2) and neglecting δ term products as second order finds:

$$\begin{aligned} J\ddot{\theta} &= -\left[\left(c - \frac{1}{2}\delta c\right)\left(\dot{x} + l\,\dot{\theta} - \dot{x}_F\right) + \left(k - \frac{1}{2}\delta k\right)\left(x + l\,\theta - x_F\right)\right]\left(l - \delta l\right) \\ &\quad + \left[\left(c + \frac{1}{2}\delta c\right)\left(\dot{x} - l\,\dot{\theta} - \dot{x}_F\right) + \left(k + \frac{1}{2}\delta k\right)\left(x - l\,\theta - x_F\right)\right]\left(l + \delta l\right) \end{aligned} \tag{10.5.1-8}$$
$$= -2c\,l^2\dot{\theta} - 2k\,l^2\theta + l\,\delta c\left(\dot{x} - \dot{x}_F\right) + l\,\delta k\left(x - x_F\right) + 2c\,\delta l\left(\dot{x} - \dot{x}_F\right) + 2k\,\delta l\left(x - x_F\right)$$

or upon rearrangement:

$$J\ddot{\theta} + 2c\,l^2\dot{\theta} + 2k\,l^2\theta = \left(l\,\delta c + 2c\,\delta l\right)\left(\dot{x} - \dot{x}_F\right) + \left(l\,\delta k + 2k\,\delta l\right)\left(x - x_F\right) \tag{10.5.1-9}$$

To combine (10.5.1-7) and (10.5.1-9), it is advantageous to use their Laplace transform equivalents:

$$\left(m\,S^2 + 2\,c\,S + 2\,k\right) X(S) \;=\; (2\,c\,S + 2\,k)\,X_F(S) + \left(-\,m\,\delta l\,S^2 + \delta c\,l\,S + \delta k\,l\right)\vartheta(S) \tag{10.5.1-10}$$

$$\left(J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2\right)\vartheta(S) \;=\; \left[\left(l\,\delta c + 2\,c\,\delta l\right)S + \;l\,\delta k + 2\,k\,\delta l\right]\left(X(S) - X_F(S)\right) \tag{10.5.1-11}$$

where

S = Laplace transform parameter.

$X(S)$, $\vartheta(S)$, $X_F(S)$ = Laplace transforms of $x(t)$, $\theta(t)$ and $x_F(t)$.

The system input is generally specified as an acceleration rather than a position vibration. We can also specify the linear response of the sensor assembly as an acceleration rather than position displacement through the following definitions and Laplace transform equivalencies:

$$\begin{aligned} a_F &\equiv \ddot{x}_F \qquad & X_F(S) &= \frac{1}{S^2}\,A_F(S) \\ a &\equiv \ddot{x} \qquad & X(S) &= \frac{1}{S^2}\,A(S) \end{aligned} \tag{10.5.1-12}$$

where

a, a_F = Acceleration associated with x, x_F.

$A(S)$, $A_F(S)$ = Laplace transforms of a, a_F.

Substituting (10.5.1-12) into (10.5.1-10) and rearranging then obtains for $A(S)$:

$$A(S) \;=\; \frac{(2\,c\,S + 2\,k)\,A_F(S) + \left(-\,m\,\delta l\,S^2 + \delta c\,l\,S + \delta k\,l\right)S^2\,\vartheta(S)}{m\,S^2 + 2\,c\,S + 2\,k} \tag{10.5.1-13}$$

Equation (10.5.1-10) can be manipulated to develop an expression for the $X(S) - X_F(S)$ term in (10.5.1-11). Dividing (10.5.1-10) by $m\,S^2 + 2\,c\,S + 2\,k$, subtracting $X_F(S)$ from both sides of the result with factorization, rearrangement, and substitution of (10.5.1-12) yields:

$$\begin{aligned} X(S) - X_F(S) \;&=\; \left(\frac{2\,c\,S + 2\,k}{m\,S^2 + 2\,c\,S + 2\,k} - 1\right)X_F(S) + \frac{\left(-\,m\,\delta l\,S^2 + \delta c\,l\,S + \delta k\,l\right)\vartheta(S)}{m\,S^2 + 2\,c\,S + 2\,k} \\ &=\; \frac{-\,m\,S^2\,X_F(S) + \left(-\,m\,\delta l\,S^2 + \delta c\,l\,S + \delta k\,l\right)\vartheta(S)}{m\,S^2 + 2\,c\,S + 2\,k} \\ &=\; \frac{-\,m\,A_F(S) + \left(-\,m\,\delta l\,S^2 + \delta c\,l\,S + \delta k\,l\right)\vartheta(S)}{m\,S^2 + 2\,c\,S + 2\,k} \end{aligned} \tag{10.5.1-14}$$

Substituting (10.5.1-14) into (10.5.1-11) with factorization and neglecting products of δ terms as second order then obtains:

$$\left(J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2\right)\vartheta(S) \;=$$

$$\left[\left(l\,\delta c + 2\,c\,\delta l\right)S + l\,\delta k + 2\,k\,\delta l\right]\frac{\left[-\,m\,A_F(S) + \left(-\,m\,\delta l\,S^2 + \delta c\,l\,S + \delta k\,l\right)\vartheta(S)\right]}{m\,S^2 + 2\,c\,S + 2\,k} \qquad (10.5.1\text{-}15)$$

$$\approx\; -\,\frac{\left[\left(l\,\delta c + 2\,c\,\delta l\right)S + l\,\delta k + 2\,k\,\delta l\right]m\,A_F(S)}{m\,S^2 + 2\,c\,S + 2\,k}$$

or, solving for $\vartheta(S)$:

$$\vartheta(S) \;=\; -\,\frac{m\left[\left(l\,\delta c + 2\,c\,\delta l\right)S + l\,\delta k + 2\,k\,\delta l\right]}{\left(J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2\right)\left(m\,S^2 + 2\,c\,S + 2\,k\right)}\,A_F(S) \qquad (10.5.1\text{-}16)$$

From (10.5.1-16) we see as expected that $\vartheta(S)$ is proportional to the δ terms. Thus, the $\vartheta(S)$ term in (10.5.1-13) multiplied by its δ term coefficient becomes second order, hence, negligible. The result is the following simplified version of (10.5.1-13) for A(S):

$$A(S) \;=\; \frac{(2\,c\,S + 2\,k)}{m\,S^2 + 2\,c\,S + 2\,k}\,A_F(S) \qquad (10.5.1\text{-}17)$$

Equations (10.5.1-16) and (10.5.1-17) describe the angular and linear response of the sensor assembly ($\vartheta(S)$ and A(S)) as a function of the applied system input acceleration $A_F(S)$. In order to cast these equations into a more recognizable form, we now introduce the following normalized error parameters and second order linear dynamic system parameter definitions (as in Reference 7 - Section 6.2.3) based on the coefficient groupings in (10.5.1-16) and (10.5.1-17):

$$\begin{aligned}
\omega_x &\equiv \sqrt{\frac{2\,k}{m}} & \qquad \zeta_x &\equiv \frac{c}{m\,\omega_x} \\
\omega_\theta &\equiv \sqrt{\frac{2\,k\,l^2}{J}} & \qquad \zeta_\theta &\equiv \frac{c\,l^2}{J\,\omega_\theta} \\
\varepsilon_k &\equiv \frac{\delta k}{k} & \qquad \varepsilon_c &\equiv \frac{\delta c}{c} \\
L &\equiv 2\,l & \qquad \varepsilon_l &\equiv \frac{\delta l}{L}
\end{aligned} \qquad (10.5.1\text{-}18)$$

where

ω_x, ζ_x = Undamped natural frequency and damping ratio for the linear vibration motion dynamic response characteristic.

$\omega_\theta, \zeta_\theta$ = Undamped natural frequency and damping ratio for the rotary vibration motion dynamic response characteristic.

L = Distance between upper and lower isolators (See Figure 10.5.1-1).

$\varepsilon_k, \varepsilon_c, \varepsilon_l$ = Normalized δk, δc, δl parameters as fractions of k, c, L.

Using the (10.5.1-18) definitions, individual terms in (10.5.1-16) and (10.5.1-17) become:

$$2\,c\,S + 2\,k = m\left(2\,\frac{c}{m}\,S + \frac{2\,k}{m}\right) = m\left(2\,\zeta_x\,\omega_x\,S + \omega_x^2\right)$$

$$m\,S^2 + 2\,c\,S + 2\,k = m\left(S^2 + 2\,\frac{c}{m}\,S + \frac{2\,k}{m}\right) = m\left(S^2 + 2\,\zeta_x\,\omega_x\,S + \omega_x^2\right)$$

$$J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2 = J\left(S^2 + 2\,\frac{c\,l^2}{J}\,S + \frac{2\,k\,l^2}{J}\right) = J\left(S^2 + 2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2\right) \qquad (10.5.1\text{-}19)$$

$$\left(l\,\delta c + 2\,c\,\delta l\right)S + l\,\delta k + 2\,k\,\delta l = c\,l\left(\frac{\delta c}{c} + 4\,\frac{\delta l}{2\,l}\right)S + k\,l\left(\frac{\delta k}{k} + 4\,\frac{\delta l}{2\,l}\right)$$
$$= \left[2\,\frac{c\,l^2}{J}\left(\varepsilon_c + 4\,\varepsilon_l\right)S + \frac{2\,k\,l^2}{J}\left(\varepsilon_k + 4\,\varepsilon_l\right)\right]\frac{J}{2\,l}$$
$$= \left[2\,\zeta_\theta\,\omega_\theta\left(\varepsilon_c + 4\,\varepsilon_l\right)S + \omega_\theta^2\left(\varepsilon_k + 4\,\varepsilon_l\right)\right]\frac{J}{L}$$

Substituting (10.5.1-19) into (10.5.1-16) - (10.5.1-17) then obtains the equivalent dynamic response relations in more familiar dynamic notation:

$$A(S) = \frac{2\,\zeta_x\,\omega_x\,S + \omega_x^2}{S^2 + 2\,\zeta_x\,\omega_x\,S + \omega_x^2}\,A_F(S)$$
$$\vartheta(S) = -\,\frac{1}{L}\,\frac{\left[2\,\zeta_\theta\,\omega_\theta\left(\varepsilon_c + 4\,\varepsilon_l\right)S + \omega_\theta^2\left(\varepsilon_k + 4\,\varepsilon_l\right)\right]}{\left(S^2 + 2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2\right)\left(S^2 + 2\,\zeta_x\,\omega_x\,S + \omega_x^2\right)}\,A_F(S) \qquad (10.5.1\text{-}20)$$

Equations (10.5.1-20) are now in a convenient form for defining the sensor assembly angular and linear motion amplitude and phase responses to sinusoidal input acceleration a_F. As in Section 10.2.1, Equation (10.2.1-2), we first identify the transfer functions to a_F input from (10.5.1-20) as:

$$H_A(S) = \frac{2\,\zeta_x\,\omega_x\,S + \omega_x^2}{S^2 + 2\,\zeta_x\,\omega_x\,S + \omega_x^2}$$

$$H_\vartheta(S) = -\frac{1}{L}\,\frac{\left[2\,\zeta_\theta\,\omega_\theta\left(\varepsilon_c + 4\,\varepsilon_l\right)S + \omega_\theta^2\left(\varepsilon_k + 4\,\varepsilon_l\right)\right]}{\left(S^2 + 2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2\right)\left(S^2 + 2\,\zeta_x\,\omega_x\,S + \omega_x^2\right)} \tag{10.5.1-21}$$

where

$H_A(S)$, $H_\vartheta(S)$ = Transfer functions relating the $A(S)$, $\vartheta(S)$ response to the $A_F(S)$ input.

As in Section 10.2.1, we then substitute $j\omega$ for S in $H_A(S)$, $H_\vartheta(S)$ and group real and imaginary terms (recognizing that j squared equals minus one by definition):

$$H_A(j\omega) = \frac{\omega_x^2 + j\,2\,\zeta_x\,\omega_x\,\omega}{\omega_x^2 - \omega^2 + j\,2\,\zeta_x\,\omega_x\,\omega}$$

$$H_\vartheta(j\omega) = -\frac{1}{L}\,\frac{\omega_\theta^2\left(\varepsilon_k + 4\,\varepsilon_l\right) + j\,2\,\zeta_\theta\,\omega_\theta\left(\varepsilon_c + 4\,\varepsilon_l\right)\omega}{\left(\omega_\theta^2 - \omega^2 + j\,2\,\zeta_\theta\,\omega_\theta\,\omega\right)\left(\omega_x^2 - \omega^2 + j\,2\,\zeta_x\,\omega_x\,\omega\right)} \tag{10.5.1-22}$$

Using generalized Equation (10.2.1-31) as a template, we identify the individual numerator and denominator polynomials in (10.5.1-22) as:

$$H_{Num_{A1}}(j\omega) = \omega_x^2 + j\,2\,\zeta_x\,\omega_x\,\omega$$

$$H_{Den_{A1}}(j\omega) = \omega_x^2 - \omega^2 + j\,2\,\zeta_x\,\omega_x\,\omega$$

$$H_{Num_{\vartheta 1}}(j\omega) = -\,\omega_\theta^2\left(\varepsilon_k + 4\,\varepsilon_l\right) - j\,2\,\zeta_\theta\,\omega_\theta\left(\varepsilon_c + 4\,\varepsilon_l\right)\omega \tag{10.5.1-23}$$

$$H_{Den_{\vartheta 1}}(j\omega) = L\left(\omega_\theta^2 - \omega^2 + j\,2\,\zeta_\theta\,\omega_\theta\,\omega\right)$$

$$H_{Den_{\vartheta 2}}(j\omega) = \omega_x^2 - \omega^2 + j\,2\,\zeta_x\,\omega_x\,\omega$$

where

$H_{Num_{Ai}}(j\omega)$, $H_{Den_{Ai}}(j\omega)$, $H_{Num_{\vartheta i}}(j\omega)$, $H_{Den_{\vartheta i}}(j\omega)$ = Numerator and denominator polynomials for $H_A(j\omega)$, $H_\vartheta(j\omega)$.

Note in (10.5.1-23) that we have associated the minus sign in the (10.5.1-22) $H_\vartheta(j\omega)$ expression with $H_{Num_{\vartheta 1}}(j\omega)$, and the $\frac{1}{L}$ multiplier in $H_\vartheta(j\omega)$ with $H_{Den_{\vartheta 1}}(j\omega)$.

From generalized Equation (10.2.1-32), we then convert (10.5.1-23) to the equivalent amplitude-ratio/phase-angle form:

$$B_{Num_{A1}}(\omega) = \sqrt{\omega_x^4 + 4\,\zeta_x^2\,\omega_x^2\,\omega^2} \qquad \phi_{Num_{A1}}(\omega) = \tan^{-1}\frac{2\,\zeta_x\,\omega}{\omega_x}$$

$$B_{Den_{A1}}(\omega) = \sqrt{\left(\omega_x^2 - \omega^2\right)^2 + 4\,\zeta_x^2\,\omega_x^2\,\omega^2} \qquad \phi_{Den_{A1}}(\omega) = \tan^{-1}\frac{2\,\zeta_x\,\omega_x\,\omega}{\omega_x^2 - \omega^2}$$

$$B_{Num_{\vartheta 1}}(\omega) = \sqrt{\omega_\theta^4\left(\varepsilon_k + 4\,\varepsilon_l\right)^2 + 4\,\zeta_\theta^2\,\omega_\theta^2\left(\varepsilon_c + 4\,\varepsilon_l\right)^2\omega^2}$$

$$\phi_{Num_{\vartheta 1}}(\omega) = \pi + \tan^{-1}\frac{2\,\zeta_\theta\left(\varepsilon_c + 4\,\varepsilon_l\right)\omega}{\omega_\theta\left(\varepsilon_k + 4\,\varepsilon_l\right)} \tag{10.5.1-24}$$

$$B_{Den_{\vartheta 1}}(\omega) = L\,\sqrt{\left(\omega_\theta^2 - \omega^2\right)^2 + 4\,\zeta_\theta^2\,\omega_\theta^2\,\omega^2}$$

$$\phi_{Den_{\vartheta 1}}(\omega) = \tan^{-1}\frac{2\,\zeta_\theta\,\omega_\theta\,\omega}{\omega_\theta^2 - \omega^2}$$

$$B_{Den_{\vartheta 2}}(\omega) = B_{Den_{A1}}(\omega) \qquad \phi_{Den_{\vartheta 2}}(\omega) = \phi_{Den_{A1}}(\omega)$$

where

$B_{Num_{A1}}(\omega)$, $\phi_{Num_{A1}}(\omega)$, $B_{Den_{A1}}(\omega)$, $\phi_{Den_{A1}}(\omega)$ = Amplitude ratios and phase angles associated with $H_{Num_{A1}}(j\omega)$, $H_{Den_{A1}}(j\omega)$.

$B_{Num_{\vartheta 1}}(\omega)$, $\phi_{Num_{\vartheta 1}}(\omega)$, $B_{Den_{\vartheta 1}}(\omega)$, $\phi_{Den_{\vartheta 1}}(\omega)$, $B_{Den_{\vartheta 2}}(\omega)$, $\phi_{Den_{\vartheta 2}}(\omega)$ = Amplitude ratios and phase angles associated with $H_{Num_{\vartheta 1}}(j\omega)$, $H_{Den_{\vartheta 1}}(j\omega)$, $H_{Den_{\vartheta 2}}(j\omega)$.

The π offset in the (10.5.1-24) $\phi_{Num_{\vartheta 1}}$ expression accounts for the negative real and negative imaginary components in $H_{Num_{\vartheta 1}}$ of Equations (10.5.1-23) which places $\phi_{Num_{\vartheta 1}}$ in the "third quadrant" of the real-imaginary plane.

Finally, we use (10.5.1-24) in generalized Equations (10.2.1-34) to obtain the amplitude ratios and phase angles for the $H_A(j\omega)$, $H_\vartheta(j\omega)$ transfer functions:

$$B_A(\omega) = \frac{B_{Num_{A1}}(\omega)}{B_{Den_{A1}}(\omega)} \qquad \phi_A(\omega) = \phi_{Num_{A1}}(\omega) - \phi_{Den_{A1}}(\omega)$$

$$B_\vartheta(\omega) = \frac{B_{Num_{\vartheta 1}}(\omega)}{B_{Den_{\vartheta 1}}(\omega)\, B_{Den_{\vartheta 2}}(\omega)} \qquad \phi_\vartheta(\omega) = \phi_{Num_{\vartheta 1}}(\omega) - \phi_{Den_{\vartheta 1}}(\omega) - \phi_{Den_{\vartheta 2}}(\omega) \tag{10.5.1-25}$$

where

$B_A(\omega)$, $\phi_A(\omega)$, $B_\vartheta(\omega)$, $\phi_\vartheta(\omega)$ = Amplitude ratios and phase angles associated with $H_A(j\omega)$, $H_\vartheta(j\omega)$.

Equations (10.5.1-25) with (10.5.1-24) define the amplitude ratio and phase angle dynamic response of the Figure 10.5.1-1 sensor assembly to applied linear sinusoidal vibrations at frequency ω, as a function of the sensor assembly isolation system dynamics, imbalances in the isolator characteristics, and center of mass offset from the nominal CG balance point. The $B_A(\omega)$, $\phi_A(\omega)$, $B_\vartheta(\omega)$, $\phi_\vartheta(\omega)$ expressions in (10.5.1-25) represent the $B(\omega)$, $\phi(\omega)$ amplitude-ratio/phase-angle terms in the Sections 10.3 and 10.4 results summarized in Equations (10.3-20) and (10.4.1-10). The input forcing function $a_F(t)$ for the (10.5.1-25) dynamic response would correspond to the $p_{Vib}(t)$ vibration source input to (10.3-20) and (10.4.1-10).

10.5.2 DYNAMIC MODEL RESPONSE TO ROTARY SYSTEM FORCING FUNCTION

Our simplified dynamic model with rotary system input forcing is based on the Figure 10.5.2-1 sketch illustrating the response of the INS sensor assembly to prescribed rotary input motion of the INS mount.

If we compare Figure 10.5.2-1 with the linear forcing model of Figure 10.5.1-1 we see that they differ in the "wall" mount interface motion. In Figure 10.5.1-1, the wall translated from left to right without rotation. In Figure 10.5.2-1, the wall has rotary motion with no net translation, causing the upper and lower sensor assembly mount interfaces to undergo differential linear left to right motion. In other respects, Figures 10.5.1-1 and 10.5.2-1 are identical. Based on the Figure 10.5.2-1 model and assuming small angle response, we write the classical Newtonian equations relating the sensor assembly angular acceleration to the applied

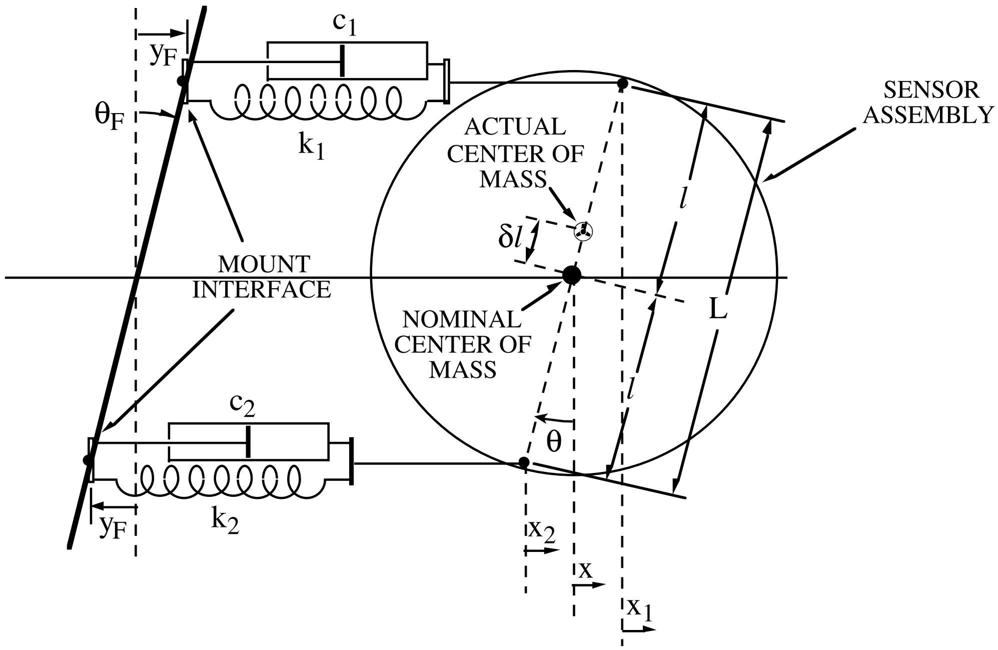


Figure 10.5.2-1 Sensor Assembly Dynamic Response To Rotary Forcing

torques around the center of mass, and the sensor assembly center of mass acceleration to the applied forces:

$$m \frac{d^2(x + \theta\, \delta l)}{dt^2} = - c_1 \left(\dot{x}_1 - \dot{y}_F\right) - k_1 \left(x_1 - y_F\right) - c_2 \left(\dot{x}_2 + \dot{y}_F\right) - k_2 \left(x_2 + y_F\right) \qquad (10.5.2\text{-}1)$$

$$J\, \ddot{\theta} = - \left[c_1 \left(\dot{x}_1 - \dot{y}_F\right) + k_1 \left(x_1 - y_F\right)\right] \left(l - \delta l\right) + \left[c_2 \left(\dot{x}_2 + \dot{y}_F\right) + k_2 \left(x_2 + y_F\right)\right] \left(l + \delta l\right) \qquad (10.5.2\text{-}2)$$

with:

$$y_F = l\, \theta_F \qquad x_1 = x + l\, \theta \qquad x_2 = x - l\, \theta \qquad (10.5.2\text{-}3)$$

where

θ_F = Angular movement of the sensor mount wall from its neutral angular orientation under forced "F" angular input motion of the sensor mount interface wall (assumed small). The neutral angular orientation is defined as the orientation of the sensor mount interface wall under zero dynamic wall motion.

y_F = Linear movement of the top and bottom isolator wall mount attachment points from their neutral position generated by θ_F. The isolator wall mount attachment point neutral positions are defined as the position of the attachment points under zero dynamic motion of the sensor mount interface wall.

Combining (10.5.2-3) finds:

$$x_1 - y_F = x + l\left(\theta - \theta_F\right) \qquad x_2 + y_F = x - l\left(\theta - \theta_F\right) \tag{10.5.2-4}$$

As in Equations (10.5.1-4) and (10.5.1-5), we define :

$$\begin{aligned} k &\equiv \frac{1}{2}\left(k_1 + k_2\right) \qquad & c &\equiv \frac{1}{2}\left(c_1 + c_2\right) \\ \delta k &\equiv k_2 - k_1 \qquad & \delta c &\equiv c_2 - c_1 \end{aligned} \tag{10.5.2-5}$$

for which we have:

$$\begin{aligned} k_1 &= k - \frac{1}{2}\delta k \qquad & k_2 &= k + \frac{1}{2}\delta k \\ c_1 &= c - \frac{1}{2}\delta c \qquad & c_2 &= c + \frac{1}{2}\delta c \end{aligned} \tag{10.5.2-6}$$

Substituting (10.5.2-4) and (10.5.2-6) into (10.5.2-1) then gives:

$$\begin{aligned} m\left(\ddot{x} + \ddot{\theta}\,\delta l\right) = &- \left(c - \frac{1}{2}\delta c\right)\left(\dot{x} + l\,(\dot{\theta} - \dot{\theta}_F)\right) - \left(k - \frac{1}{2}\delta k\right)\left[x + l\,(\theta - \theta_F)\right] \\ &- \left(c + \frac{1}{2}\delta c\right)\left(\dot{x} - l\,(\dot{\theta} - \dot{\theta}_F)\right) - \left(k + \frac{1}{2}\delta k\right)\left(x - l\,(\theta - \theta_F)\right) \\ = &- 2\,c\,\dot{x} - 2\,k\,x + \delta c\, l\left(\dot{\theta} - \dot{\theta}_F\right) + \delta k\, l\left(\theta - \theta_F\right) \end{aligned} \tag{10.5.2-7}$$

or with rearrangement and grouping:

$$m\,\ddot{x} + 2\,c\,\dot{x} + 2\,k\,x = -\,m\,\delta l\,\ddot{\theta} + \delta c\,l\,(\dot{\theta} - \dot{\theta}_F) + \delta k\,l\,(\theta - \theta_F) \tag{10.5.2-8}$$

Making the same substitutions in (10.5.2-2) while neglecting δ products as second order obtains:

$$\begin{aligned} J\,\ddot{\theta} = &- \left[\left(c - \frac{1}{2}\delta c\right)\left(\dot{x} + l\,(\dot{\theta} - \dot{\theta}_F)\right) + \left(k - \frac{1}{2}\delta k\right)\left(x + l\,(\theta - \theta_F)\right)\right]\left(l - \delta l\right) \\ &+ \left[\left(c + \frac{1}{2}\delta c\right)\left(\dot{x} - l\,(\dot{\theta} - \dot{\theta}_F)\right) + \left(k + \frac{1}{2}\delta k\right)\left(x - l\,(\theta - \theta_F)\right)\right]\left(l + \delta l\right) \\ = &- 2\,c\,l^2\,(\dot{\theta} - \dot{\theta}_F) - 2\,k\,l^2\,(\theta - \theta_F) + \delta c\,l\,\dot{x} + \delta k\,l\,x + 2\,c\,\delta l\,\dot{x} + 2\,k\,\delta l\,x \end{aligned} \tag{10.5.2-9}$$

or after rearrangement and grouping:

$$J\ddot{\theta} + 2\,c\,l^2\,\dot{\theta} + 2\,k\,l^2\theta$$
$$= 2\,c\,l^2\,\dot{\theta}_F + 2\,k\,l^2\,\theta_F + (\delta c\,l + 2\,c\,\delta l)\,\dot{x} + (\delta k\,l + 2\,k\,\delta l)\,x \tag{10.5.2-10}$$

If we take the Laplace transform of Equation (10.5.2-10) and add/subtract $J\,S^2\,\vartheta_F(s)$ on the right, we can obtain with rearrangement, a solution for $\vartheta(S) - \vartheta_F(S)$ as a function of $\vartheta_F(S)$ and X(S). Taking the Laplace transform of Equation (10.5.2-8) and adding/subtracting $m\,\delta l\,S^2\,\vartheta_F(S)$ on the right, would yield with rearrangement, a solution for X(S) as a function of $\vartheta_F(S)$ and $\vartheta(S) - \vartheta_F(S)$. By then substituting the previous $\vartheta(S) - \vartheta_F(S)$ solution (from (10.5.2-10)) in the (10.5.2-8) X(S) result and rearranging, we can find a solution for X(S) as a function of $\vartheta_F(S)$. This procedure would show that x is on the order of $l\,\theta_F$ multiplied by δ terms. Therefore, the x products with the delta terms in (10.5.2-10) are second order, and (10.5.2-10) reduces to:

$$J\ddot{\theta} + 2\,c\,l^2\,\dot{\theta} + 2\,k\,l^2\,\theta = 2\,c\,l^2\,\dot{\theta}_F + 2\,k\,l^2\,\theta_F \tag{10.5.2-11}$$

The Laplace transform of (10.5.2-11) is:

$$\left(J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2\right)\vartheta(S) = \left(2\,c\,l^2\,S + 2\,k\,l^2\right)\vartheta_F(S) \tag{10.5.2-12}$$

where

$\vartheta_F(S)$ = Laplace transform of θ_F.

from which we obtain for $\vartheta(S)$:

$$\vartheta(S) = \frac{2\,c\,l^2\,S + 2\,k\,l^2}{J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2}\,\vartheta_F(S) \tag{10.5.2-13}$$

Returning to Equation (10.5.2-8) for x, the Laplace transform gives:

$$\left(m\,S^2 + 2\,c\,S + 2\,k\right)X(S) = -\,m\,\delta l\,S^2\,\vartheta(S) + \left(\delta c\,l\,S + \delta k\,l\right)\left(\vartheta(S) - \vartheta_F(S)\right) \tag{10.5.2-14}$$

The $\left(\vartheta(S) - \vartheta_F(S)\right)$ term in (10.5.2-14) is from (10.5.2-13):

$$\vartheta(S) - \vartheta_F(S) = \frac{-\,J\,S^2}{J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2}\,\vartheta_F(S) \tag{10.5.2-15}$$

Substituting (10.5.2-13) and (10.5.2-15) into (10.5.2-14) then yields:

$$\left(m\,S^2 + 2\,c\,S + 2\,k\right) X(S) \;=\; - \frac{m\,\delta l\left(2\,c\,l^2\;S + 2\,k\,l^2\right) + \left(\delta c\;l\,S + \delta k\;l\right) J}{J\,S^2 + 2\,c\,l^2\;S + 2\,k\,l^2}\,S^2\;\vartheta_F(S) \tag{10.5.2-16}$$

or after rearrangement and use of (10.5.1-12) to represent linear motion response as an acceleration:

$$A(S) \;=\; - \frac{\left[m\,\delta l\left(2\,c\,l^2\;S + 2\,k\,l^2\right) + \left(\delta c\;l\,S + \delta k\;l\right) J\right] S^4}{\left(J\,S^2 + 2\,c\,l^2\;S + 2\,k\,l^2\right)\left(m\,S^2 + 2\,c\,S + 2\,k\right)}\,\vartheta_F(S) \tag{10.5.2-17}$$

The (10.5.2-13) and (10.5.2-17) results can be expressed in more familiar terms by incorporating the (10.5.1-18) definitions for which particular terms become:

$$\begin{aligned}
&2\,c\,l^2\;S + 2\,k\,l^2 \;=\; J\left(2\,\frac{c\,l^2}{J}\,S + \frac{2\,k\,l^2}{J}\right) \;=\; J\left(2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2\right)\\
&J\,S^2 + 2\,c\,l^2\,S + 2\,k\,l^2 \;=\; J\left(S^2 + 2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2\right)\\
&m\,\delta l\left(2\,c\,l^2\;S + 2\,k\,l^2\right) + \left(\delta c\;l\,S + \delta k\;l\right) J\\
&\qquad =\; m\,J\,2l\left(2\,\frac{c\,l^2}{J}\,S + \frac{2\,k\,l^2}{J}\right)\frac{\delta l}{2\,l} + m\,J\,l\left(\frac{c}{m}\,\frac{\delta c}{c}\,S + \frac{1}{2}\,\frac{2\,k}{m}\,\frac{\delta k}{k}\right)\\
&\qquad =\; m\,J\,2l\left[\left(2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2\right)\varepsilon_l + \frac{1}{2}\left(\zeta_x\;\omega_x\;\varepsilon_c\,S + \frac{1}{2}\,\omega_x^2\;\varepsilon_k\right)\right]\\
&\qquad =\; m\,J\,L\left[\left(2\,\zeta_\theta\,\omega_\theta\,\varepsilon_l + \frac{1}{2}\,\zeta_x\;\omega_x\;\varepsilon_c\right) S + \omega_\theta^2\,\varepsilon_l + \frac{1}{4}\,\omega_x^2\;\varepsilon_k\right]\\
&m\,S^2 + 2\,c\,S + 2\,k \;=\; m\left(S^2 + 2\,\zeta_x\;\omega_x\;S + \omega_x^2\right)
\end{aligned} \tag{10.5.2-18}$$

from which (10.5.2-13) and (10.5.2-17) become:

$$\begin{aligned}
&\vartheta(s) \;=\; \frac{2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2}{S^2 + 2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2}\,\vartheta_F(s)\\
&A(S) \;=\; -\,L\,\frac{\left[\left(2\,\zeta_\theta\,\omega_\theta\,\varepsilon_l + \frac{1}{2}\,\zeta_x\;\omega_x\;\varepsilon_c\right) S + \omega_\theta^2\,\varepsilon_l + \frac{1}{4}\,\omega_x^2\;\varepsilon_k\right] S^4}{\left(S^2 + 2\,\zeta_\theta\,\omega_\theta\,S + \omega_\theta^2\right)\left(S^2 + 2\,\zeta_x\;\omega_x\;S + \omega_x^2\right)}\,\vartheta_F(S)
\end{aligned} \tag{10.5.2-19}$$

As in Section 10.5.1, the Equation (10.5.2-19) results for $\vartheta(S)$, $A(S)$ can be converted to the equivalent amplitude-ratio/phase-angle response to $\vartheta_F(S)$ sinusoidal inputs using the Section 10.2.1 conversion process (left as an exercise for the curious reader). We note, however, that vibration effects analysis is usually used to predict INS performance under specified vibration inputs. Typical INS test specifications only dictate vibration performance evaluation under prescribed linear acceleration vibration inputs (i.e., a_F in Section 10.5.1). Performance testing under specified angular vibrations is generally not required.

10.6 VIBRATION EFFECTS ANALYSIS SIMULATION PROGRAM

Sections 10.3 and 10.4 derived analytical formulas for calculating strapdown INS performance parameters as a function of input vibration and the dynamic response of the physical structure connecting the vibration source to the INS sensor assembly. Section 10.5 described simplified models for the structural dynamic characteristics. If the summary equations in these sections are reviewed (i.e., Equations (10.3-20), (10.4.1-10), (10.4.1-11), (10.4.2-44), (10.5.1-24), (10.5.1-25) and (10.5.2-19)), it should be apparent that the results lend themselves nicely to programming into a computer software simulation program for numerical vibration effects analysis. This section outlines the basic equations that might be incorporated in such a simulation. For compatibility with typical INS test specifications, we will only address the structure of a simulation program for analyzing INS performance under prescribed linear acceleration input in a fixed direction in the B Frame. Thus, the Section 10.3 and 10.4 results we use will be based on $p_{Vib}(t)$ being an input vibration acceleration for which the Section 10.5.1 analytical model applies with $p_{Vib}(t)$ identified as $a_F(t)$.

This section is divided into three parts, Sections 10.6.1, 10.6.2 and 10.6.3. Section 10.6.1 describes a simulation program for evaluating the attitude/velocity/position response of the sensor assembly to sinusoidal and random vibration inputs. Due to their development complexity, separate sections are provided (10.6.2 and 10.6.3) to describe the portion of the simulation for evaluating the position algorithm error $\delta \underline{R}_{SF/Algo}(t)$ response to input sinusoidal vibration inputs, and the response to sinusoidal and random vibrations of the sculling error due to sensor dynamics $\delta \dot{v}_{SF/Scul/SnsDyn_z}$.

10.6.1 SIMULATION PROGRAM FOR ATTITUDE/VELOCITY/POSITION VIBRATION RESPONSE ANALYSIS

Section 10.5.1 defines the strapdown sensor assembly linear and angular amplitude-ratio/phase-angle response to linear excitation for the simplified Figure 10.5.1-1 sensor assembly mounting arrangement. Let us use these results to define the $\dot{v}_{SF/Scul_z}$ term in (10.3-20) (repeated below) for our simulation program to be based on the Figure 10.5.1-1 vibration model.

$$\dot{v}_{SF/Scul_z} = \frac{1}{2}\, p_{Vib0}^2 \Big[B_{\theta_x} B_{a_{SF_y}} \cos\left(\phi_{\theta_x} - \phi_{a_{SF_y}}\right) - B_{\theta_y} B_{a_{SF_x}} \cos\left(\phi_{\theta_y} - \phi_{a_{SF_x}}\right) \Big] \tag{10.6.1-1}$$

For reference purposes, recall that $\dot{v}_{SF/Scul_z}$ was derived from Equation (10.1.2.1-13) and as an extrapolation from (10.3-11) based on the (10.3-8) sensor assembly B Frame motion model. We select the rotation axis in Figure 10.5.1-1 as representing the X axis for the Equation (10.6.1-1) $\dot{v}_{SF/Scul_z}$ B Frame angular rate, and the acceleration input axis in Figure 10.5.1-1 as representing the Y axis for the $\dot{v}_{SF/Scul_z}$ B Frame acceleration. Note that defining Y as the vibration axis differs from the Figure 10.5.1-1 notation in which the linear vibration axis input and linear response are identified as X. We also note that our Figure 10.5.1-1 model has a linear response along only the acceleration input axis, hence, the acceleration response perpendicular to this axis is zero. Thus we can write the equivalency between (10.6.1-1) and the (10.5.1-24) - (10.5.1-25) amplitude-ratio/phase-angle response terms for a particular input vibration frequency Ω as:

$$\begin{aligned} B_{\theta_x} &= B_{\vartheta}(\Omega) & \phi_{\theta_x} &= \phi_{\vartheta}(\Omega) \\ B_{a_{SF_y}} &= B_A(\Omega) & \phi_{a_{SF_y}} &= \phi_A(\Omega) \\ B_{a_{SF_x}} &= 0 & B_{\theta_y} &= 0 \end{aligned} \tag{10.6.1-2}$$

Using (10.6.1-2), we see from (10.5.1-25) that the $\phi_{\theta_x} - \phi_{a_{SF_y}}$ term in (10.6.1-1) is:

$$\begin{aligned} \phi_{\theta_x} - \phi_{a_{SF_y}} &= \phi_{\vartheta}(\Omega) - \phi_A(\Omega) \\ &= \phi_{Num_{\vartheta 1}}(\Omega) - \phi_{Den_{\vartheta 1}}(\Omega) - \phi_{Den_{\vartheta 2}}(\Omega) - \phi_{Num_{A1}}(\Omega) + \phi_{Den_{A1}}(\Omega) \end{aligned} \tag{10.6.1-3}$$

But, from (10.5.1-24):

$$\phi_{Den_{A1}}(\Omega) = \phi_{Den_{\vartheta 2}}(\Omega) \tag{10.6.1-4}$$

Thus, (10.6.1-3) simplifies to:

$$\phi_{\theta_x} - \phi_{a_{SF_y}} = \phi_{Num_{\vartheta 1}}(\Omega) - \phi_{Den_{\vartheta 1}}(\Omega) - \phi_{Num_{A1}}(\Omega) \tag{10.6.1-5}$$

or with (10.5.1-24) for the individual phase angle components:

$$\phi_{\theta_x} - \phi_{a_{SF_y}} = \pi + \tan^{-1} \frac{2\,\zeta_\theta \left(\varepsilon_c + 4\,\varepsilon_l\right)\Omega}{\omega_\theta \left(\varepsilon_k + 4\,\varepsilon_l\right)} - \tan^{-1} \frac{2\,\zeta_\theta\, \omega_\theta\, \Omega}{\omega_\theta^2 - \Omega^2} - \tan^{-1} \frac{2\,\zeta_y\, \Omega}{\omega_y} \tag{10.6.1-6}$$

where

ζ_y, ω_y = Damping ratio and undamped natural frequency associated with the linear Y axis response of the sensor assembly to Y axis linear input vibration. Note that the equivalent parameters in Equation (10.5.1-24) are identified as ζ_x, ω_x to correspond with the X axis input vibration axis definition in Figure 10.5.1-1.

From Equation (10.6.1-6) we see that $\phi_{\theta_x} - \phi_{a_{SF_y}}$ starts at π for $\Omega = 0$ (corresponding to a maximum magnitude for the cos $\left(\phi_{\theta_x} - \phi_{a_{SF_y}}\right)$ term in (10.6.1-1)), and then traces a complex history as Ω increases from 0, depending on the values for the sensor assembly mount imbalance and dynamic response characteristics. As a worst case analysis and to simplify the simulation program being developed, we can approximate the magnitude of cos $(\phi_{\theta_x} - \phi_{a_{SF_y}})$ in (10.6.1-1) as unity over all Ω. While doing this, we also note that Equation (10.6.1-6) is based on the simplified Figure 10.5.1-1, which only very approximately represents the response of the actual sensor assembly, with the largest uncertainties usually occurring in the phase angle response estimates. This further justifies the previous approximation, allowing that the actual phase response may have a worse impact on (10.6.1-1) than predicted by the Figure 10.5.1-1 model. Based on these arguments, Equation (10.6.1-1) with (10.6.1-2) simplifies to the following order of magnitude expression for our simulation program:

$$\dot{v}_{SF/Scul_z} \approx \frac{1}{2} B_\vartheta(\Omega)\, B_A(\Omega)\, a^2_{Vib_0} \tag{10.6.1-7}$$

where

$a_{Vib_0} = p_{Vib_0}$ now identified as the input sinusoidal acceleration amplitude.

with, from (10.5.1-24) - (10.5.1-25):

$$\begin{aligned} B_A(\Omega) &= \sqrt{\frac{\omega_y^4 + 4\,\zeta_y^2\,\omega_y^2\,\Omega^2}{\left(\omega_y^2 - \Omega^2\right)^2 + 4\,\zeta_y^2\,\omega_y^2\,\Omega^2}} \\ B_\vartheta(\Omega) &= \frac{1}{L}\sqrt{\frac{\omega_\theta^4\left(\varepsilon_k + 4\,\varepsilon_l\right)^2 + 4\,\zeta_\theta^2\,\omega_\theta^2\left(\varepsilon_c + 4\,\varepsilon_l\right)^2\Omega^2}{\left[\left(\omega_\theta^2 - \Omega^2\right)^2 + 4\,\zeta_\theta^2\,\omega_\theta^2\,\Omega^2\right]\left[\left(\omega_y^2 - \Omega^2\right)^2 + 4\,\zeta_y^2\,\omega_y^2\,\Omega^2\right]}} \end{aligned} \tag{10.6.1-8}$$

with ζ_y and ω_y as defined earlier. For a worst case estimate, the ε parameter values used in (10.6.1-8) can all be set to have the same polarity.

Equations (10.6.1-7) - (10.6.1-8) are based on sculling along axis Z generated by angular vibration around axis X coupled with linear vibration along axis Y, the X angular vibration being the result of sensor assembly mounting imbalances (as in Figure 10.5.1-1) driven by the Y linear vibration. However, in a three-dimensional world, mounting imbalances would also produce angular vibrations around axis Z in response to the same linear vibration along Y. For example, consider that the Figure 10.5.1-1 sensor assembly is cylindrical in shape with the cylinder axis normal to the page (and to the applied acceleration), and that the top and bottom isolators consist of two parallel sets, one set on each end of the cylinder. Consider that the cylinder center-of-mass is nominally on the cylinder axis midway between the isolator mounts, and that the actual center-of-mass may be displaced from the nominal along the cylinder axis and (as shown in Figure 10.5.1-1), perpendicular to the cylinder axis. The sensor assembly will then have an angular response in addition to that shown in Figure 10.5.1-1, that is normal to the applied acceleration, about an axis parallel to the "wall" mount. The additional angular response around the axis parallel to the "wall" mount (called axis Z for this sculling discussion) coupled with the Y axis linear vibration, produces sculling along axis X.

Let us define the isolators on one end of the cylinder as set "a" and those on the other end as set "b". The spring/damping coefficients associated with these can then be identified as k_{1a}, c_{1a}, k_{1b}, c_{1b} and k_{2a}, c_{2a}, k_{2b}, c_{2b} in which the "1" and "2" notation is the same as in Figure 10.5.1-1 denoting the "top" and "bottom" isolators. Using Equations (10.5.1-4) as a model and the coordinate definitions in the previous paragraph, the net spring/damping coefficients for the sensor assembly along the linear vibration input axis Y will be $k_y = \frac{1}{2}\left(k_{2a} + k_{2b} + k_{1a} + k_{1b}\right)$ and $c_y = \frac{1}{2}\left(c_{2a} + c_{2b} + c_{1a} + c_{1b}\right)$ with the "y" subscript designating the resulting Y axis linear response. Based on the previous coordinate definitions, angular vibration around axis X will be generated by imbalances between the bottom ("2") and top ("1") isolators, or, as in (10.5.1-4), by $\delta k_x = \left(k_{2a} + k_{2b} - k_{1a} - k_{1b}\right)$ and $\delta c_x = \left(c_{2a} + c_{2b} - c_{1a} - c_{1b}\right)$ with the "x" subscript designating the resulting X axis angular response. Angular vibrations around Z will be generated by imbalances between the "a" and "b" isolators, or by $\delta k_z = \left(k_{1b} + k_{2b} - k_{1a} - k_{2a}\right)$ and $\delta c_z = \left(c_{1b} + c_{2b} - c_{1a} - c_{2a}\right)$ with the "z" subscript designating the resulting Z axis angular response. Similarly, under applied Y axis linear vibration, mass imbalance along axis Z (δl_z) will also produce angular vibration about X (as depicted in Figure 10.5.1-1), and mass imbalance along X (δl_x) will generate angular vibration around Z.

If we assume equivalent dynamic characteristics and sensor assembly imbalances for Z and X axis angular response to Y axis linear vibration (i.e., $\delta k_x = \delta k_z = \delta k$, $\delta c_x = \delta c_z = \delta c$, $\delta l_x = \delta l_z = \delta l$ and (10.5.1-18) for ε_k, ε_c, ε_l using $k = k_y$), we can easily write the X axis sculling equivalent to (10.6.1-7) for our simulation program as:

$$\dot{v}_{SF/Scul_x} \approx \frac{1}{2} B_{\vartheta}(\Omega)\, B_A(\Omega)\, a_{Vib_0}^2 \tag{10.6.1-9}$$

where

$\dot{v}_{SF/Scul_x}$ = Sculling along sensor assembly axis X generated by angular vibration around axis Z caused by sensor assembly mounting asymmetries under Y axis linear vibration.

with $B_{\vartheta}(\Omega)$ and $B_A(\Omega)$ as given in (10.6.1-8).

Let's apply a similar treatment to the $\delta\dot{v}_{SF/Algo\text{-}m_z}$ term in (10.3-20) repeated below:

$$\begin{aligned} \delta\dot{v}_{SF/Algo\text{-}m_z} = \frac{1}{2} p_{Vib_0}^2 \Big[& B_{\theta_x} B_{a_{SF_y}} \cos\left(\phi_{\theta_x} - \phi_{a_{SF_y}}\right) \\ & - B_{\theta_y} B_{a_{SF_x}} \cos\left(\phi_{\theta_y} - \phi_{a_{SF_x}}\right)\Big] \left\{\left[1 + \frac{1}{3}\left(1 - \cos \Omega\, T_l\right)\right]\left(\frac{\sin \Omega\, T_l}{\Omega\, T_l}\right) - 1\right\} \end{aligned} \tag{10.6.1-10}$$

To find the worst case magnitude of $\delta\dot{v}_{SF/Algo\text{-}m_z}$ in (10.6.1-10), we set $\cos\left(\phi_{\theta_x} - \phi_{a_{SF_y}}\right)$ equal to unity with sign corresponding to the sign of the $\left\{\left[1 + \frac{1}{3}\left(1 - \cos \Omega\, T_l\right)\right]\left(\frac{\sin \Omega\, T_l}{\Omega\, T_l}\right) - 1\right\}$ term. Using (10.6.1-2) in (10.6.1-10), this is equivalent to:

$$\delta\dot{v}_{SF/Algo\text{-}m_z} \approx \frac{1}{2} B_{\vartheta}(\Omega)\, B_A(\Omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos \Omega\, T_l\right)\right]\left(\frac{\sin \Omega\, T_l}{\Omega\, T_l}\right) - 1 \right| a_{Vib_0}^2 \tag{10.6.1-11}$$

with $B_{\vartheta}(\omega)$, $B_A(\omega)$ from (10.6.1-8).

As in (10.6.1-9), we can also identify an X axis sculling algorithm error under the Y axis input vibration:

$$\delta\dot{v}_{SF/Algo\text{-}m_x} \approx \frac{1}{2} B_{\vartheta}(\Omega)\, B_A(\Omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos \Omega\, T_l\right)\right]\left(\frac{\sin \Omega\, T_l}{\Omega\, T_l}\right) - 1 \right| a_{Vib_0}^2 \tag{10.6.1-12}$$

where

$\delta\dot{v}_{SF/Algo\text{-}m_x}$ = Sculling algorithm error along sensor assembly axis X generated by angular vibration around axis Z due to sensor assembly mounting asymmetries under Y axis linear vibration.

Applying the same technique for coning rate analysis in our simulation program, we first write for the coning rate from $\dot{\Phi}_{Con_z}$ in (10.3-20) using the revised coordinate frame definitions:

$$\dot{\Phi}_{Con_y} = \frac{1}{2}\,\Omega\,B_{\theta_x}\,B_{\theta_z}\,p^2_{Vib\,0}\,\sin\left(\phi_{\theta_x} - \phi_{\theta_z}\right) \qquad (10.6.1\text{-}13)$$

where

$\dot{\Phi}_{Con_y}$ = Coning rate around axis Y (the linear vibration input axis) generated by the angular response around axes X and Z produced by sensor assembly imbalances.

B_{θ_z}, ϕ_{θ_z} = Amplitude ratio and phase angle relating input sinusoidal vibration inputs to the angular response around sensor assembly axis Z.

For reference purposes, recall that $\dot{\Phi}_{Con_z}$ in (10.3-20) (repeated above as $\dot{\Phi}_{Con_y}$ for the new coordinate definitions) was derived from Equation (10.1.1.1-13) and as an extrapolation from (10.3-5) based on the (10.3-4) sensor assembly B Frame motion model. We note that in Figure 10.5.1-1, the angular response shown is around an axis that is normal to the applied acceleration and normal to the plane of the page. Now consider as in the sculling discussion preceding (10.6.1-9), that the Figure 10.5.1-1 linear vibration will also produce a second angular vibration that is normal to the applied acceleration, but about an axis parallel to the "wall" mount. The amplitude-ratio/phase-angle response for this second angular response (B_{θ_z} and ϕ_{θ_z}) will be of the same form as Equations (10.5.1-24) - (10.5.1-25), but with parameter values corresponding to the other axis. We selected the rotation axis shown in Figure 10.5.1-1 as representing the X axis for the Equation (10.6.1-13) $\dot{\Phi}_{Con_y}$ B Frame angular rate, and the "second" rotation axis (parallel to the Figure 10.5.1-1 wall mount) as representing the Z axis for the $\dot{\Phi}_{Con_y}$ B Frame angular rate. For simplicity, let us also assume (as in the sculling discussion) that the sensor assembly nominal angular vibration modes about the previously two defined angular vibration axes have identical dynamic response characteristics (i.e., the same ζ_θ and ω_θ). Then we can write an equivalency between the (10.6.1-13) and (10.5.1-24) - (10.5.1-25) amplitude-ratio/phase-angle response terms for a particular input frequency Ω as:

$$B_{\theta_x} = \frac{B_{Num/\vartheta 1_x}(\Omega)}{B_{Den/\vartheta 1_x}(\Omega)\;B_{Den/\vartheta 2_x}(\Omega)} \qquad B_{\theta_z} = \frac{B_{Num/\vartheta 1_z}(\Omega)}{B_{Den/\vartheta 1_z}(\Omega)\;B_{Den/\vartheta 2_z}(\Omega)}$$

$$\phi_{\theta_x} = \phi_{Num/\vartheta 1_x}(\Omega) - \phi_{Den/\vartheta 1_x}(\Omega) - \phi_{Den/\vartheta 2_x}(\Omega) \qquad (10.6.1\text{-}14)$$

$$\phi_{\theta_z} = \phi_{Num/\vartheta 1_z}(\Omega) - \phi_{Den/\vartheta 1_z}(\Omega) - \phi_{Den/\vartheta 2_z}(\Omega)$$

$$B_{Num/\vartheta 1_x}(\Omega) = \sqrt{\omega_\theta^4 \left(\varepsilon_{k_x} + 4\,\varepsilon_{l_x}\right)^2 + 4\,\zeta_\theta^2\,\omega_\theta^2 \left(\varepsilon_{c_x} + 4\,\varepsilon_{l_x}\right)^2 \Omega^2}$$

$$B_{Num/\vartheta 1_z}(\Omega) = \sqrt{\omega_\theta^4 \left(\varepsilon_{k_z} + 4\,\varepsilon_{l_z}\right)^2 + 4\,\zeta_\theta^2\,\omega_\theta^2 \left(\varepsilon_{c_z} + 4\,\varepsilon_{l_z}\right)^2 \Omega^2}$$

$$\phi_{Num/\vartheta 1_x}(\Omega) = \pi + \tan^{-1} \frac{2\,\zeta_\theta \left(\varepsilon_{c_x} + 4\,\varepsilon_{l_x}\right) \Omega}{\omega_\theta \left(\varepsilon_{k_x} + 4\,\varepsilon_{l_x}\right)}$$

$$\phi_{Num/\vartheta 1_z}(\Omega) = \pi + \tan^{-1} \frac{2\,\zeta_\theta \left(\varepsilon_{c_z} + 4\,\varepsilon_{l_z}\right) \Omega}{\omega_\theta \left(\varepsilon_{k_z} + 4\,\varepsilon_{l_z}\right)} \qquad (10.6.1\text{-}15)$$

$$B_{Den/\vartheta 1_x}(\Omega) = B_{Den/\vartheta 1_z}(\Omega) = L \sqrt{\left(\omega_\theta^2 - \Omega^2\right)^2 + 4\,\zeta_\theta^2\,\omega_\theta^2\,\Omega^2}$$

$$\phi_{Den/\vartheta 1_x}(\Omega) = \phi_{Den/\vartheta 1_z}(\Omega) = \tan^{-1} \frac{2\,\zeta_\theta\,\omega_\theta\,\Omega}{\omega_\theta^2 - \Omega^2}$$

$$B_{Den/\vartheta 2_x}(\Omega) = B_{Den/\vartheta 2_z}(\Omega) = \sqrt{\left(\omega_y^2 - \Omega^2\right)^2 + 4\,\zeta_y^2\,\omega_y^2\,\Omega^2}$$

$$\phi_{Den/\vartheta 2_x}(\Omega) = \phi_{Den/\vartheta 2_z}(\Omega) = \tan^{-1} \frac{2\,\zeta_y\,\omega_y\,\Omega}{\omega_y^2 - \Omega^2}$$

where

$\varepsilon_{k_x}, \varepsilon_{c_x}, \varepsilon_{l_x}$ = Values for the ε_k, ε_c, ε_l imbalances associated with the angular response shown in Figure 10.5.1-1 (about the newly above defined X axis). Note that ε_{l_x} is caused by center-of-mass offset shown in Figure 10.5.1-1 along the newly above defined Z axis (parallel to the Figure 10.5.1-1 wall mount).

$\varepsilon_{k_z}, \varepsilon_{c_z}, \varepsilon_{l_z}$ = Values for the ε_k, ε_c, ε_l imbalances associated with the additional angular response (not shown in Figure 10.5.1-1) about the newly above defined Z axis (parallel to the Figure 10.5.1-1 wall mount). Note that ε_{l_z} is caused by center-of-mass offset (not shown in Figure 10.5.1-1) along the newly above defined X axis (a center-of-mass offset along the cylinder axis perpendicular to the page of Figure 10.5.1-1).

$B_{Num/\vartheta 1_x}(\Omega)$, $B_{Num/\vartheta 1_z}(\Omega)$, $\phi_{Num/\vartheta 1_x}(\Omega)$, $\phi_{Num/\vartheta 1_z}(\Omega)$, $B_{Den/\vartheta 1_x}(\Omega)$, $B_{Den/\vartheta 1_z}(\Omega) \cdots$
= Values for $B_{Num_{\vartheta 1}}(\Omega)$, $\phi_{Num_{\vartheta 1}}(\Omega)$, $B_{Den_{\vartheta 1}}(\Omega) \cdots$ associated with the B Frame X, Z axis angular response.

From (10.6.1-14) and (10.6.1-15), the $\phi_{\theta_x} - \phi_{\theta_z}$ term in (10.6.1-13) is:

$$\phi_{\theta_x} - \phi_{\theta_z} = \phi_{Num/\vartheta 1_x}(\Omega) - \phi_{Num/\vartheta 1_z}(\Omega)$$
$$= \tan^{-1}\frac{2\,\zeta_\theta\left(\varepsilon_{c_x} + 4\,\varepsilon_{l_x}\right)\Omega}{\omega_\theta\left(\varepsilon_{k_x} + 4\,\varepsilon_{l_x}\right)} - \tan^{-1}\frac{2\,\zeta_\theta\left(\varepsilon_{c_z} + 4\,\varepsilon_{l_z}\right)\Omega}{\omega_\theta\left(\varepsilon_{k_z} + 4\,\varepsilon_{l_z}\right)} \tag{10.6.1-16}$$

We see from (10.6.1-16) that $\phi_{\theta_x} - \phi_{\theta_z}$ is zero for zero Ω and then varies with Ω depending on the particular values for the X, Z sensor assembly imbalances. From Equation (10.6.1-13) we see that the worst case value for $\phi_{\theta_x} - \phi_{\theta_z}$ is $\frac{\pi}{2}$ for which the sine term is one and $\dot{\Phi}_{Con_y}$ is maximum. Since the phase response in (10.6.1-16) is only a rough approximation of the actual phase response, and since the sensor assembly imbalances vary between axes and sensor assemblies, it is reasonable to approximate the phase as equal to $\frac{\pi}{2}$ for a worst case estimate of $\dot{\Phi}_{Con_y}$. Note that the $\phi_{\theta_x} - \phi_{\theta_z}$ equal to $\frac{\pi}{2}$ approximation will always be incorrect near $\Omega = 0$, however, we see from (10.6.1-13) that the error is nullified by the Ω multiplier that sets $\dot{\Phi}_{Con_y}$ equal to zero for $\Omega = 0$. We can also assume from an amplitude ratio standpoint that the X, Z imbalances are of the same magnitude so that we can write:

$$B_{\theta_x} = B_{\theta_z} = B_\vartheta(\Omega) \tag{10.6.1-17}$$

with $B_\vartheta(\Omega)$ as given in (10.6.1-8). Using the previous approximations, the worst case analysis form of (10.6.1-13) becomes simply:

$$\dot{\Phi}_{Con_y} \approx \frac{1}{2}\,\Omega\,B_\vartheta^2(\Omega)\;a_{Vib_0}^2 \tag{10.6.1-18}$$

Using the same procedure but choosing the sign of $\phi_{\theta_x} - \phi_{\theta_z}$ to match the net sign of the Ω terms, the worst case response for $\dot{\beta}_{m_z}$ in (10.3-20) would be (using the new coordinate axis definitions):

$$\dot{\beta}_{m_y} \approx \frac{1}{2}\,\Omega\,B_\vartheta^2(\Omega)\left|1 - \frac{\sin\,\Omega T_m}{\Omega T_m}\right| a_{Vib_0}^2 \tag{10.6.1-19}$$

where

$\dot{\beta}_{m_y}$ = Sensor assembly Y axis component of $\dot{\underline{\beta}}_m$ as defined following Equation (10.1.1.2.1-14).

It is also of interest to have the simulation program evaluate the magnitudes of the analytical model sensor assembly angular and acceleration responses. We can use the generic $\overline{f(t)}$ in Equations (10.3-20) for this purpose which, from (10.3-20) and (10.1.4.1-3), is:

$$f(t) = h_x(t)\, h_y(t)$$
$$\overline{f(t)} = \frac{1}{2} B_{h_x} B_{h_y} \cos\left(\phi_{h_x} - \phi_{h_y}\right) p^2_{Vib_0} \tag{10.6.1-20}$$

For the angular and acceleration responses we choose the mean squared values as the performance parameters of interest. Then, using the revised coordinate definitions, the equivalency between these and the f(t) generic parameter in (10.6.1-20) is:

For Acceleration Response:

$$f(t) = a_{SF_y}(t)^2 \qquad h_x(t) = h_y(t) = a_{SF_y}(t)$$
$$\phi_{h_x} = \phi_{h_y} = \phi_A \qquad B_{h_x} = B_{h_y} = B_A$$

For Angular Response:

$$f(t) = \theta_{z\text{ or }x}(t)^2 \qquad h_x(t) = h_y(t) = \theta_{z\text{ or }x}(t)$$
$$\phi_{h_x} = \phi_{h_y} = \phi_\vartheta \qquad B_{h_x} = B_{h_y} = B_\vartheta \tag{10.6.1-21}$$

where

$a_{SF_y}(t)$ = Linear acceleration response of the sensor assembly along the Y axis due to Y axis acceleration inputs.

$\theta_{z\text{ or }x}(t)$ = Angular (integrated angular rate) response of the sensor assembly around the Z or X axis due to Y axis acceleration inputs and sensor assembly imbalance.

Identifying Ω frequency dependence and assuming comparable angular response around the Z and X axes, we see from (10.6.1-20) and (10.6.1-21) that:

$$\overline{a_{SF_y}(t)^2} = \frac{1}{2} B_A^2(\Omega)\, a^2_{Vib_0} \qquad \overline{\theta_{z\text{ or }x}(t)^2} = \frac{1}{2} B_\vartheta^2(\Omega)\, a^2_{Vib_0} \tag{10.6.1-22}$$

where

$\overline{a_{SF_y}(t)^2}, \overline{\theta_{z\,or\,x}(t)^2}$ = Mean squared values for the sensor assembly acceleration $a_{SF_y}(t)$ and angular $\theta_{z\,or\,x}(t)$ response to the applied sinusoidal input acceleration vibration.

with $B_\vartheta(\Omega)$, $B_A(\Omega)$ provided from Equations (10.6.1-8).

The procedures leading to (10.6.1-7), (10.6.1-11), (10.6.1-18) and (10.6.1-19) can also be used for the remaining terms in Equations (10.3-20) (with the exception of $\delta\underline{R}_{SF/Algo}(t)$ and $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ which are treated separately in Sections 10.6.2 and 10.6.3). Except for $\delta\underline{R}_{SF/Algo}(t)$ and $\delta\dot{v}_{SF/Scul/SnsDyn_z}$, the following is a summary of the (10.3-20) terms based on worst case analysis using (10.6.1-8) for $B_\vartheta(\Omega)$ and $B_A(\Omega)$. For worst case analysis, the ε terms in (10.6.1-8) can be set to have the same polarity.

$$\dot{\Phi}_{Con_y} = \frac{1}{2}\,\Omega\, B_\vartheta^2(\Omega)\; a_{Vib_0}^2$$

$$\dot{\beta}_{m_y} = \frac{1}{2}\,\Omega\, B_\vartheta^2(\Omega) \left| 1 - \frac{\sin \Omega T_m}{\Omega T_m} \right| a_{Vib_0}^2$$

$$\dot{\beta}_{Algo\text{-}m\,y} = \frac{1}{2}\,\Omega\, B_\vartheta^2(\Omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos\, \Omega\, T_l\right)\right] \left(\frac{\sin\, \Omega\, T_l}{\Omega\, T_l} \right) - \frac{\sin\, \Omega\, T_m}{\Omega\, T_m} \right| a_{Vib_0}^2$$

$$\delta\dot{\Phi}_{Algo\text{-}m\,y} = \delta\dot{\beta}_{Algo\text{-}m\,y} = \frac{1}{2}\,\Omega\, B_\vartheta^2(\Omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos\, \Omega\, T_l\right)\right] \left(\frac{\sin\, \Omega\, T_l}{\Omega\, T_l} \right) - 1 \right| a_{Vib_0}^2$$

$$\dot{v}_{SF/Scul_{z\,or\,x}} = \frac{1}{2}\, B_\vartheta(\Omega)\, B_A(\Omega)\; a_{Vib_0}^2 \qquad (10.6.1\text{-}23)$$

$$\Delta\dot{v}_{Scul\text{-}m_{z\,or\,x}} = \frac{1}{2}\, B_\vartheta(\Omega)\, B_A(\Omega) \left| 1 - \frac{\sin \Omega T_m}{\Omega T_m} \right| a_{Vib_0}^2$$

(Continued)

$$\Delta\dot{v}_{Scul/Algo\text{-}m_{z\,or\,x}} = \frac{1}{2} B_{\vartheta}(\Omega)\, B_A(\Omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos \Omega T_l\right)\right] \left(\frac{\sin \Omega T_l}{\Omega T_l}\right) - \frac{\sin \Omega T_m}{\Omega T_m} \right| a_{Vib_0}^2$$

$$\delta\dot{v}_{SF/Algo\text{-}m_{z\,or\,x}} = \delta\Delta\dot{v}_{Scul/Algo\text{-}m_{z\,or\,x}} = \frac{1}{2} B_{\vartheta}(\Omega)\, B_A(\Omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos \Omega T_l\right)\right] \left(\frac{\sin \Omega T_l}{\Omega T_l}\right) - 1 \right| a_{Vib_0}^2$$

$$\overline{\delta a_{Accl_{G2}}} = L_{G2}\, \frac{1}{4}\, B_A^2(\Omega)\, a_{Vib_0}^2 \qquad \text{(10.6.1-23) (Continued)}$$

$$\overline{\delta a_{Accl_{Aniso}}} = L_{Aniso}\, \frac{1}{2}\, \Omega^2\, B_{\vartheta}^2(\Omega)\, a_{Vib_0}^2$$

$$\overline{a_{SF_y}(t)^2} = \frac{1}{2} B_A^2(\Omega)\, a_{Vib_0}^2$$

$$\overline{\theta_{z\,or\,x}(t)^2} = \frac{1}{2} B_{\vartheta}^2(\Omega)\, a_{Vib_0}^2$$

where

$(\ )_{z\ or\ x}$ = Terms generated by angular vibrations around the sensor assembly Z and X axes due to mounting imbalances under Y axis linear vibration. The same equation applies for the Z or X axis response.

The $\frac{1}{4}$ term in the (10.6.1-23) $\overline{\delta a_{Accl_{G2}}}$ expression arises based on considering the sensor assembly response to vibration input as being along an axis in the B Frame X, Y plane that has equal components along X, Y (i.e., at 45 degrees to X and Y). This produces an acceleration amplitude p_{Vib_0} in the (10.3-20) $\overline{\delta a_{Accl_{G2}}}$ expression of $\frac{1}{\sqrt{2}}\, a_{Vib_0}$ along each of X and Y, which produces $\frac{1}{4}\, a_{Vib_0}^2$ in the (10.6.1-23) $\overline{\delta a_{Accl_{G2}}}$ equation. The additional Ω^2 term in the (10.6.1-23) $\overline{\delta a_{Accl_{Aniso}}}$ expression arises because the B amplitude ratio terms for $\overline{\delta a_{Accl_{Aniso}}}$ in (10.3-20) represent angular rate responses, while $B_{\vartheta}(\Omega)$ in (10.6.1-8) is the angle response. The angular rate response is the derivative of the angle response, hence, the angular rate based transfer function equals S times the angle based transfer function. The associated angular rate amplitude ratio is the (10.6.1-8) $B_{\vartheta}(\Omega)$ angle response amplitude ratio multiplied by Ω.

Equations (10.6.1-23) with (10.6.1-8) can be easily programmed into a digital computer analysis program for worst case numerical evaluation of the vibration sensitive performance parameters under a prescribed vibration input acceleration amplitude a_{Vib_0} and frequency Ω. The sensor assembly mount characteristics would be additional program inputs (i.e., ω_y, ζ_y, ω_θ, ζ_θ, L, ε_k, ε_c and ε_l) that will vary for each application. Typical values used by the writer for some of these parameters for a reasonably accurate strapdown INS have been $\varepsilon_k = 0.05$, $\varepsilon_c = 0.05$, $\varepsilon_l = 0.1 \text{ inch} / L$, $\omega_y = 2\,\pi\,50 \text{ Hz}$, and $\zeta_y = 0.125$. Generally the rotary dynamic response undamped natural frequency and damping ratio (ω_θ and ζ_θ) are not readily available parameters. The writer has found that a reasonable approximation can be obtained as follows based on the (10.5.1-18) definitions for which (using the revised coordinate axis definitions):

$$\frac{\omega_\theta}{\omega_y} = \sqrt{\frac{m\,l^2}{J}} \qquad \frac{\zeta_\theta}{\zeta_y} = \frac{m\,l^2}{J}\,\frac{\omega_y}{\omega_\theta} = \sqrt{\frac{m\,l^2}{J}} \tag{10.6.1-24}$$

If the sensor assembly were a solid cylinder of radius l, its moment of inertia J about the cylinder axis would be $\frac{m\,l^2}{2}$ for which $\sqrt{\frac{m\,l^2}{J}}$ in (10.6.1-24) would be $\sqrt{2}$. Thus, a reasonable approximation for ω_θ, ζ_θ is to set them equal to $\sqrt{2}$ times the linear response equivalents ω_y, ζ_y.

Equations (10.6.1-23) are based on a worst case approximation of Equations (10.3-20) for a discrete frequency sinusoidal acceleration input. The identical methodology leading to (10.6.1-23) can also be applied to Equations (10.4.1-10) - (10.4.1-11) to determine the equivalent expressions for random acceleration vibration inputs. Equation (10.4.2-44) for the position error variance response applies directly (with $B_{aSF}(\omega) = B_A(\omega)$ and $G_{pVib}(\omega) = G_{aVib}(\omega)$). Except for $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ (treated separately in Section 10.6.3), the results are:

$$\mathcal{E}\left(\dot{\Phi}_{\mathrm{Con}_y}\right) = \int_0^\infty \omega\, B_\vartheta^2(\omega)\, G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\dot{\beta}_{\mathrm{m}_y}\right) = \int_0^\infty \omega\, B_\vartheta^2(\omega) \left| 1 - \frac{\sin \omega T_m}{\omega T_m} \right| G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\dot{\beta}_{\mathrm{Algo\text{-}m}\,y}\right) = \int_0^\infty \omega\, B_\vartheta^2(\omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos \omega T_l\right)\right] \left(\frac{\sin \omega T_l}{\omega T_l} \right) - \frac{\sin \omega T_m}{\omega T_m} \right| G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\delta\dot{\Phi}_{\mathrm{Algo\text{-}m}\,y}\right) = \mathcal{E}\left(\delta\dot{\beta}_{\mathrm{Algo\text{-}m}\,y}\right)$$

$$= \int_0^\infty \omega\, B_\vartheta^2(\omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos \omega T_l\right)\right] \left(\frac{\sin \omega T_l}{\omega T_l} \right) - 1 \right| G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\dot{v}_{\mathrm{SF/Scul}_{z\ \mathrm{or}\ x}}\right) = \int_0^\infty B_\vartheta(\omega)\, B_A(\omega)\, G_{aVib}(\omega)\, d\omega \qquad (10.6.1\text{-}25)$$

$$\mathcal{E}\left(\Delta\dot{v}_{\mathrm{Scul\text{-}m}_{z\ \mathrm{or}\ x}}\right) = \int_0^\infty B_\vartheta(\omega)\, B_A(\omega) \left| 1 - \frac{\sin \omega T_m}{\omega T_m} \right| G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\Delta\dot{v}_{\mathrm{Scul/Algo\text{-}m}_{z\ \mathrm{or}\ x}}\right) = \int_0^\infty B_\vartheta(\omega)\, B_A(\omega) \left| \left[1 \right.\right.$$

$$\left.\left. + \frac{1}{3}\left(1 - \cos \omega T_l\right)\right] \left(\frac{\sin \omega T_l}{\omega T_l} \right) - \frac{\sin \omega T_m}{\omega T_m} \right| G_{aVib}(\omega)\, d\omega$$

(Continued)

$$\mathcal{E}\left(\delta\dot{v}_{SF/Algo\text{-}m_{z\,or\,x}}\right) = \mathcal{E}\left(\delta\Delta\dot{v}_{Scul/Algo\text{-}m_{z\,or\,x}}\right)$$

$$= \int_0^{\infty} B_{\vartheta}(\omega)\, B_A(\omega) \left| \left[1 + \frac{1}{3}\left(1 - \cos\,\omega\,T_l\right)\right]\left(\frac{\sin\,\omega\,T_l}{\omega\,T_l}\right) - 1 \right| G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\delta R^2_{SF/Algo}(t)\right) = (t - t_0)^2 \int_0^{\infty} B_A^2(\omega)\, \frac{2}{\omega^2} \left\{ E(\omega)^{\,2} + \frac{1}{6}\,(\omega'\,T_l)^2 \left[E(\omega)\, f_1(\omega'\,T_l) \right.\right.$$

$$\left.\left. + \frac{1}{12}\,(\omega'\,T_l)^2\, f_2(\omega'\,T_l) \right] \right\} f_2\left(\omega'(t - t_0)\right) G_{aVib}(\omega)\, d\omega$$

$$\omega' = \omega - \frac{2\,\pi}{T_l}\left(\frac{\omega\,T_l}{2\,\pi}\right)_{Intgr} \qquad E(\omega) = \frac{f_1(\omega'\,T_l)}{2\,f_2(\omega'\,T_l)} - \frac{\omega'}{\omega}$$

f_1 and f_2 defined by Equations (10.1.3.2.3-16)

$$\mathcal{E}\left(\overline{\delta a_{Accl_{G2}}}\right) = L_{G2} \int_0^{\infty} \frac{1}{2}\, B_A^2(\omega)\, G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\overline{\delta a_{Accl_{Aniso}}}\right) = L_{Aniso} \int_0^{\infty} \omega^2\, B_{\vartheta}^2(\omega)\, G_{aVib}(\omega)\, d\omega \qquad \text{(10.6.1-25) (Continued)}$$

$$\mathcal{E}\left(\overline{a_{SF_y}(t)^2}\right) = \int_0^{\infty} B_A^2(\omega)\, G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\overline{\theta_{z\,or\,x}(t)^2}\right) = \int_0^{\infty} B_{\vartheta}^2(\omega)\, G_{aVib}(\omega)\, d\omega$$

$$\mathcal{E}\left(\overline{a_{Vib}(t)^2}\right) = \int_0^{\infty} G_{aVib}(\omega)\, d\omega$$

where

$G_{aVib}(\omega)$ = Random acceleration vibration input power spectral density.

$\mathcal{E}\left(\overline{a_{Vib}(t)^2}\right)$ = Mean squared value of random vibration acceleration input.

$B_{\vartheta}(\omega)$, $B_A(\omega)$ = Values for $B_{\vartheta}(\Omega)$, $B_A(\Omega)$ in (10.6.1-8) with $\Omega = \omega$ (and for worst case analysis, ε's set to have the same polarity).

Equations (10.6.1-25) with (10.6.1-8) can be easily programmed into the digital computer analysis program for worst case numerical evaluation of vibration sensitive performance under a prescribed random vibration input acceleration power spectral density $G_{aVib}(\omega)$ provided as a numerical function of frequency ω. The integration operations in (10.6.1-25) can be executed with any simple digital integration algorithm (e.g., trapezoidal integration).

10.6.2 DEVELOPMENT OF THE $\delta \underline{R}_{SF/Algo}(t)$ SINUSOIDAL INPUT RESPONSE FOR WORST CASE SIMULATION ANALYSIS

In this section we develop an expression for worst case simulation analysis of $\delta \underline{R}_{SF/Algo}(t)$ under sinusoidal vibration exposure (in Equations (10.3-20)) starting with the equivalent (10.4.2-4) form repeated below (using $B_A(\Omega)$ from (10.6.1-8) for B_{aSF} and a_{Vib_0} for p_{Vib_0}):

$$\begin{aligned}\delta \underline{R}_{SF/Algo}(t) = &- \underline{u}_{Vib} \frac{1}{\Omega^2} a_{Vib_0} B_A(\Omega) \left\{ \frac{\Omega}{\Omega'} \left(\frac{f_1(\Omega' T_l)}{2 f_2(\Omega' T_l)} - \frac{\Omega'}{\Omega} \right.\right. \\ &\left. + \frac{1}{12} \Omega' T_l \sin \Omega' T_l \right) \left[\sin\left(\Omega'(t - t_0) + \Omega t_0 + \psi_{pVib} + \phi_{aSF}\right) \right. \\ &\left. - \sin(\Omega t_0 + \psi_{pVib} + \phi_{aSF}) \right] - \frac{1}{12} \Omega T_l \left[\cos\left(\Omega'(t - t_0) + \Omega t_0 + \psi_{pVib} + \phi_{aSF}\right) \right. \\ &\left.\left. - \cos(\Omega t_0 + \psi_{pVib} + \phi_{aSF}) \right] (1 - \cos \Omega' T_l) \right\}\end{aligned} \qquad (10.6.2\text{-}1)$$

As in an earlier discussion, we recognize that sensor assembly phase angle response predictions using our simplified analytical models may be significantly in error, hence, we seek the phase angle response ϕ_{aSF} that will maximize $\delta \underline{R}_{SF/Algo}(t)$ as a worst case prediction. For mathematical simplicity, let us first define:

$$\begin{aligned} A &\equiv - \frac{1}{\Omega^2} a_{Vib_0} B_A(\Omega) \frac{\Omega}{\Omega'} \left(\frac{f_1(\Omega' T_l)}{2 f_2(\Omega' T_l)} - \frac{\Omega'}{\Omega} + \frac{1}{12} \Omega' T_l \sin \Omega' T_l \right) \\ B &\equiv \frac{1}{\Omega^2} a_{Vib_0} B_A(\Omega) \frac{1}{12} \Omega T_l (1 - \cos \Omega' T_l) \\ \phi &\equiv \Omega t_0 + \psi_{pVib} + \phi_{aSF} \qquad \tau \equiv t - t_0 \end{aligned} \qquad (10.6.2\text{-}2)$$

so that using (10.4.2-10), Equation (10.6.2-1) becomes the simplified form for the signed magnitude $\delta R_{SF/Algo}(\tau)$:

$$\begin{aligned}\delta R_{SF/Algo}(\tau) &= \underline{u}_{Vib} \cdot \delta \underline{R}_{SF/Algo}(\tau) \\ &= A\left(\sin(\Omega'\tau + \phi) - \sin\phi\right) + B\left(\cos(\Omega'\tau + \phi) - \cos\phi\right)\end{aligned} \tag{10.6.2-3}$$

To find the maximum (or minimum) of $\delta R_{SF/Algo}(\tau)$ with phase angle ϕ_{aSF}, we take its derivative with respect to ϕ_{aSF} (which from (10.6.2-2) is the same as the derivative with respect to ϕ), and equate the result to zero:

$$\begin{aligned}&\frac{d}{d\phi}\left[A\left(\sin(\Omega'\tau + \phi) - \sin\phi\right) + B\left(\cos(\Omega'\tau + \phi) - \cos\phi\right)\right] \\ &\qquad = A\left(\cos(\Omega'\tau + \phi) - \cos\phi\right) - B\left(\sin(\Omega'\tau + \phi) - \sin\phi\right) = 0\end{aligned} \tag{10.6.2-4}$$

From (10.6.2-4) we see that for maximum or minimum $\delta R_{SF/Algo}(\tau)$:

$$\frac{\cos(\Omega'\tau + \phi) - \cos\phi}{\sin(\Omega'\tau + \phi) - \sin\phi} = \frac{B}{A} \tag{10.6.2-5}$$

Substituting (10.6.2-5) in (10.6.2-3) then yields:

$$\begin{aligned}\delta R_{SF/Algo}(\tau) &= \left[A + \frac{\cos(\Omega'\tau + \phi) - \cos\phi}{\sin(\Omega'\tau + \phi) - \sin\phi} B\right]\left(\sin(\Omega'\tau + \phi) - \sin\phi\right) \\ &= \left(A + \frac{B^2}{A}\right)\left(\sin(\Omega'\tau + \phi) - \sin\phi\right) = \left[1 + \left(\frac{B}{A}\right)^2\right] A\left(\sin(\Omega'\tau + \phi) - \sin\phi\right)\end{aligned} \tag{10.6.2-6}$$

The sinusoidal term in (10.6.2-6) is equivalently:

$$\sin(\Omega'\tau + \phi) - \sin\phi = \sin\Omega'\tau\cos\phi - (1 - \cos\Omega'\tau)\sin\phi \tag{10.6.2-7}$$

Substituting (10.6.2-7) in (10.6.2-6) and applying the (10.1.3.2.3-16) definitions yields a singularity free version of $\delta R_{SF/Algo}(\tau)$:

$$\begin{aligned}\delta R_{SF/Algo}(\tau) &= \left[1 + \left(\frac{B\,\Omega'}{A\,\Omega'}\right)^2\right] A\,\Omega'\tau\left(\frac{\sin\Omega'\tau}{\Omega'\tau}\cos\phi - \Omega'\tau\,\frac{(1 - \cos\Omega'\tau)}{(\Omega'\tau)^2}\sin\phi\right) \\ &= \left[1 + \left(\frac{B\,\Omega'}{A\,\Omega'}\right)^2\right] A\,\Omega'\tau\left(f_1(\Omega'\tau)\cos\phi - \Omega'\tau\, f_2(\Omega'\tau)\sin\phi\right)\end{aligned} \tag{10.6.2-8}$$

The following general equalities will prove useful:

$$G = \mathrm{Sign}(G)\,|G| \qquad (\mathrm{Sign}(G))^2 = 1 \qquad G\,\mathrm{Sign}(G) = |G|$$
$$\mathrm{Sign}\left(\frac{G}{H}\right) = \frac{\mathrm{Sign}(G)}{\mathrm{Sign}(H)} = \mathrm{Sign}(G\,H) = \mathrm{Sign}(G)\,\mathrm{Sign}(H) \tag{10.6.2-9}$$

in which G and H are arbitrary parameters and where

$\mathrm{Sign}(\;) = 1$ for $(\;) \geq 0$ and -1 for $(\;) < 0$.

Using appropriate equalities from (10.6.2-9) we see that:

$$\cos\phi = \mathrm{Sign}(\cos\phi)\,|\cos\phi|$$
$$\sin\phi = \cos\phi\tan\phi = \cos\phi\,\mathrm{Sign}(\tan\phi)\,|\tan\phi| = \cos\phi\,\mathrm{Sign}(\tan\phi)\left|\frac{\sin\phi}{\cos\phi}\right| \tag{10.6.2-10}$$
$$= \mathrm{Sign}(\cos\phi)\,|\cos\phi|\,\mathrm{Sign}(\tan\phi)\,\frac{|\sin\phi|}{|\cos\phi|} = \mathrm{Sign}(\cos\phi)\,|\sin\phi|\,\mathrm{Sign}(\tan\phi)$$

With (10.6.2-10), Equation (10.6.2-8) is equivalently:

$$\delta R_{SF/Algo}(\tau) = \left[1 + \left(\frac{B\,\Omega'}{A\,\Omega'}\right)^2\right] A\,\Omega'\,\tau\,\mathrm{Sign}(\cos\phi)\Big(f_1(\Omega'\,\tau)\,|\cos\phi| - \mathrm{Sign}(\tan\phi)\,\Omega'\,\tau\,f_2(\Omega'\,\tau)\,|\sin\phi|\Big) \tag{10.6.2-11}$$

From basic trigonometry:

$$|\sin\phi| = \frac{1}{\sqrt{1 + \mathrm{ctn}^2\phi}} \qquad |\cos\phi| = \frac{1}{\sqrt{1 + \tan^2\phi}} \tag{10.6.2-12}$$

with which (10.6.2-11) becomes for the magnitude of the maximum or minimum $\delta R_{SF/Algo}(\tau)$:

$$|\delta R_{SF/Algo}(\tau)| = \left[1 + \left(\frac{B\,\Omega'}{A\,\Omega'}\right)^2\right] \left|A\,\Omega'\,\tau\left(\frac{f_1(\Omega'\,\tau)}{\sqrt{1 + \tan^2\phi}} - \mathrm{Sign}(\tan\phi)\,\Omega'\,\tau\,\frac{f_2(\Omega'\,\tau)}{\sqrt{1 + \mathrm{ctn}^2\phi}}\right)\right| \tag{10.6.2-13}$$

The tangent and cotangent functions in Equation (10.6.2-13) can be determined by rearranging (10.6.2-4) as follows:

$$
\begin{aligned}
A \cos(\Omega' \tau + \phi) - A \cos \phi - B \sin(\Omega' \tau + \phi) + B \sin \phi &= 0 \\
= - A (1 - \cos \Omega' \tau) \cos \phi - A \sin \Omega' \tau \sin \phi & \\
- B \sin \Omega' \tau \cos \phi + B (1 - \cos \Omega' \tau) \sin \phi &
\end{aligned} \tag{10.6.2-14}
$$

or

$$
\left[B (1 - \cos \Omega' \tau) - A \sin \Omega' \tau \right] \sin \phi - \left[A (1 - \cos \Omega' \tau) + B \sin \Omega' \tau \right] \cos \phi = 0 \tag{10.6.2-15}
$$

Applying the (10.1.3.2.3-16) formulas, Equation (10.6.2-15) shows that for the maximum absolute $\delta R_{SF/Algo}(\tau)$, the tan ϕ term in (10.6.2-13) is given by:

$$
\begin{aligned}
\tan \phi &= \frac{\sin \phi}{\cos \phi} = \frac{A (1 - \cos \Omega' \tau) + B \sin \Omega' \tau}{B (1 - \cos \Omega' \tau) - A \sin \Omega' \tau} \\
&= \frac{A \dfrac{(\Omega' \tau)^2}{(\Omega' \tau)^2} (1 - \cos \Omega' \tau) + B \dfrac{\Omega' \tau}{\Omega' \tau} \sin \Omega' \tau}{B \dfrac{(\Omega' \tau)^2}{(\Omega' \tau)^2} (1 - \cos \Omega' \tau) - A \dfrac{\Omega' \tau}{\Omega' \tau} \sin \Omega' \tau} \\
&= \frac{A (\Omega' \tau)^2 f_2(\Omega' \tau) + B \Omega' \tau f_1(\Omega' \tau)}{B (\Omega' \tau)^2 f_2(\Omega' \tau) - A \Omega' \tau f_1(\Omega' \tau)}
\end{aligned} \tag{10.6.2-16}
$$

or, after dividing by - A $\Omega' \tau$ and rearranging the denominator:

$$
\tan \phi = \frac{- \Omega' \tau f_2(\Omega' \tau) - \dfrac{B \Omega'}{A \Omega'} f_1(\Omega' \tau)}{f_1(\Omega' \tau) - \dfrac{B \Omega'}{A \Omega'} \Omega' \tau f_2(\Omega' \tau)} \tag{10.6.2-17}
$$

The ctn ϕ term in (10.6.2-13) is the reciprocal of (10.6.2-17):

$$
\operatorname{ctn} \phi = \frac{f_1(\Omega' \tau) - \dfrac{B \Omega'}{A \Omega'} \Omega' \tau f_2(\Omega' \tau)}{- \Omega' \tau f_2(\Omega' \tau) - \dfrac{B \Omega'}{A \Omega'} f_1(\Omega' \tau)} \tag{10.6.2-18}
$$

Using the second row formulas in (10.6.2-9), the Sign(tan ϕ) term in (10.6.2-13) can be calculated from (10.6.2-17) as:

$$\chi \equiv \left(- \Omega' \tau \, f_2(\Omega' \tau) - \frac{B\,\Omega'}{A\,\Omega'} f_1(\Omega' \tau) \right) \left(f_1(\Omega' \tau) - \frac{B\,\Omega'}{A\,\Omega'} \Omega' \tau \, f_2(\Omega' \tau) \right) \qquad (10.6.2\text{-}19)$$
$$\text{Sign}(\tan \phi) = \text{Sign}(\chi)$$

Lastly, the $A\,\Omega'$ and $\dfrac{B\,\Omega'}{A\,\Omega'}$ terms in (10.6.2-13) and (10.6.2-17) - (10.6.2-19) are from (10.6.2-2) using the (10.1.3.2.3-16) f_1 definition for $\sin \Omega' T_l$:

$$\frac{B\,\Omega'}{A\,\Omega'} = -\frac{1}{12} \frac{\Omega' T_l \, (1 - \cos \Omega' T_l)}{F(\Omega)} \qquad A\,\Omega' = -\frac{1}{\Omega} B_A(\Omega)\, F(\Omega)\, a_{Vib_0}$$
$$F(\Omega) \equiv \frac{f_1(\Omega' T_l)}{2\, f_2(\Omega' T_l)} - \frac{\Omega'}{\Omega} + \frac{1}{12} (\Omega' T_l)^2 \, f_1(\Omega' T_l) \qquad (10.6.2\text{-}20)$$

Summarizing, the maximum $\left| \delta R_{SF/Algo}(t) \right|$ is provided by Equation (10.6.2-13) with (10.6.2-17) - (10.6.2-20) and with τ as defined in (10.6.2-2). In combination, the overall result for addition to our simulation program discrete sinusoidal vibration performance parameter list in Equations (10.6.1-23) is:

$$\left| \delta R_{SF/Algo}(t) \right| = (t - t_0) \frac{1}{\Omega} B_A(\Omega) \left[1 + \frac{1}{144} \left(\frac{\Omega' T_l \, (1 - \cos \Omega' T_l)}{F(\Omega)} \right)^2 \right] \left| F(\Omega) \right| \left| \frac{f_1\big(\Omega'(t - t_0)\big)}{\sqrt{1 + \tan^2 \phi}} - \text{Sign}(\chi)\, \Omega'(t - t_0) \frac{f_2\big(\Omega'(t - t_0)\big)}{\sqrt{1 + \text{ctn}^2 \phi}} \right| a_{Vib_0}$$

$$F(\Omega) \equiv \frac{f_1(\Omega' T_l)}{2\, f_2(\Omega' T_l)} - \frac{\Omega'}{\Omega} + \frac{1}{12} (\Omega' T_l)^2 \, f_1(\Omega' T_l)$$

$$\tan \phi = \frac{- \Omega'(t - t_0)\, f_2\big(\Omega'(t - t_0)\big) + \dfrac{1}{12} \dfrac{\Omega' T_l \, (1 - \cos \Omega' T_l)}{F(\Omega)} f_1\big(\Omega'(t - t_0)\big)}{f_1\big(\Omega'(t - t_0)\big) + \dfrac{1}{12} \dfrac{\Omega' T_l \, (1 - \cos \Omega' T_l)}{F(\Omega)} \Omega'(t - t_0)\, f_2\big(\Omega'(t - t_0)\big)} \qquad (10.6.2\text{-}21)$$

$$\text{ctn}\, \phi = \frac{f_1\big(\Omega'(t - t_0)\big) + \dfrac{1}{12} \dfrac{\Omega' T_l \, (1 - \cos \Omega' T_l)}{F(\Omega)} \Omega'(t - t_0)\, f_2\big(\Omega'(t - t_0)\big)}{- \Omega'(t - t_0)\, f_2\big(\Omega'(t - t_0)\big) + \dfrac{1}{12} \dfrac{\Omega' T_l \, (1 - \cos \Omega' T_l)}{F(\Omega)} f_1\big(\Omega'(t - t_0)\big)}$$

(Continued)

$$\chi = \left(- \Omega'(t - t_0)\, f_2\big(\Omega'(t - t_0)\big) + \frac{1}{12} \frac{\Omega'\, T_l\, (1 - \cos\, \Omega'\, T_l)}{F(\Omega)}\, f_1\big(\Omega'(t - t_0)\big) \right) \Big(f_1\big(\Omega'(t - t_0)\big)$$

$$+ \frac{1}{12} \frac{\Omega'\, T_l\, (1 - \cos\, \Omega'\, T_l)}{F(\Omega)}\, \Omega'(t - t_0)\, f_2\big(\Omega'(t - t_0)\big) \qquad \text{(10.6.2-21) (Continued)}$$

10.6.3 DEVELOPMENT OF THE $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ SINUSOIDAL AND RANDOM INPUT RESPONSE FOR WORST CASE SIMULATION ANALYSIS

In this section we develop equations for worst case simulation analysis of the $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ response to sinusoidal and random vibration input exposure. We begin with the $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ expression in Equations (10.3-20) (which was duplicated from (10.3-19)) repeated below:

$$\delta\dot{v}_{SF/Scul/SnsDyn_z} = \frac{1}{2}\, p_{Vib_0}^2 \Big\{ B_{\theta_x} B_{aSF_y} \Big[B_{\omega ARS_x} B_{aAccl_y} \cos\big(\phi_{\theta_x} + \phi_{\omega ARS_x} - \phi_{aSF_y} - \phi_{aAccl_y}\big) - \cos\big(\phi_{\theta_x} - \phi_{aSF_y}\big) \Big] \qquad (10.6.3\text{-}1)$$

$$- B_{\theta_y} B_{aSF_x} \Big[B_{\omega ARS_y} B_{aAccl_x} \cos\big(\phi_{\theta_y} + \phi_{\omega ARS_y} - \phi_{aSF_x} - \phi_{aAccl_x}\big) - \cos\big(\phi_{\theta_y} - \phi_{aSF_x}\big) \Big] \Big\}$$

As in our previous discussions, we recognize that the sensor assembly phase angle response predictions using our simplified analytical models may be significantly in error, hence, we seek the phase angle response that will maximize $\delta\dot{v}_{SF/Scul/SnsDyn_z}$ as a worst case prediction. In particular, the sensor assembly phase angle response terms having the uncertainty are ϕ_{θ_x}, ϕ_{θ_y}, ϕ_{aSF_x}, ϕ_{aSF_y} which appear in (10.6.3-1) as $\phi_{\theta_x} - \phi_{aSF_y}$ and $\phi_{\theta_y} - \phi_{aSF_x}$ groupings. We assume that the inertial sensor phase angle terms $\phi_{\omega ARS_x}$, $\phi_{\omega ARS_y}$, ϕ_{aAccl_x}, ϕ_{aAccl_y} are known accurately, hence, will not be adjusted for worst case study. For mathematical simplicity, let us first define the sensor assembly phase angle terms of interest as:

$$\Delta\phi_1 \equiv \phi_{\theta_x} - \phi_{aSF_y} \qquad \Delta\phi_2 \equiv \phi_{\theta_y} - \phi_{aSF_x} \qquad (10.6.3\text{-}2)$$

Then the outer { } bracketed term in (10.6.3-1) (which we identify as Ψ) becomes:

$$\Psi = B_{\theta_x} B_{aSF_y} \Psi_1 - B_{\theta_y} B_{aSF_x} \Psi_2$$

$$\Psi_1 \equiv B_{\omega ARS_x} B_{aAccl_y} \cos\big(\Delta\phi_1 + \phi_{\omega ARS_x} - \phi_{aAccl_y}\big) - \cos \Delta\phi_1 \qquad (10.6.3\text{-}3)$$

$$\Psi_2 \equiv B_{\omega ARS_y} B_{aAccl_x} \cos\big(\Delta\phi_2 + \phi_{\omega ARS_y} - \phi_{aAccl_x}\big) - \cos \Delta\phi_2$$

$\delta\dot{v}_{SF/Scul/SnsDyn_z}$ will be maximized if we find the $\Delta\phi_1, \Delta\phi_2$ that maximizes Ψ in (10.6.3-3). The maximizing solution is determined by taking the partial derivative of Ψ with respect to $\Delta\phi_1$ and $\Delta\phi_2$ and setting them to zero. From the form of (10.6.3-3) that finds Ψ_1 and Ψ_2 independent (in terms of $\Delta\phi_1$ and $\Delta\phi_2$), we see that Ψ maximization can be accomplished by maximizing Ψ_1 and minimizing Ψ_2 individually. Thus:

$$\frac{\partial \Psi_1}{\partial \Delta\phi_1} = - B_{\omega ARS_x} B_{aAccl_y} \sin\left(\Delta\phi_1 + \phi_{\omega ARS_x} - \phi_{aAccl_y}\right) + \sin \Delta\phi_1 = 0 \qquad (10.6.3\text{-}4)$$

$$\frac{\partial \Psi_2}{\partial \Delta\phi_2} = - B_{\omega ARS_y} B_{aAccl_x} \sin\left(\Delta\phi_2 + \phi_{\omega ARS_y} - \phi_{aAccl_x}\right) + \sin \Delta\phi_2 = 0 \qquad (10.6.3\text{-}5)$$

The solution to (10.6.3-4) is found by first expanding the leading sine term:

$$\begin{aligned} \sin\left(\Delta\phi_1 + \phi_{\omega ARS_x} - \phi_{aAccl_y}\right) = {} & \sin \Delta\phi_1 \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \\ & + \cos \Delta\phi_1 \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \end{aligned} \qquad (10.6.3\text{-}6)$$

Then, substituting (10.6.3-6) in (10.6.3-4) gives:

$$\begin{aligned} & - \left[B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) - 1\right] \sin \Delta\phi_1 \\ & \quad - B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \cos \Delta\phi_1 = 0 \end{aligned} \qquad (10.6.3\text{-}7)$$

which, upon rearrangement is:

$$\begin{aligned} & B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) - 1 \\ & \quad = - B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \operatorname{ctn} \Delta\phi_1 \end{aligned} \qquad (10.6.3\text{-}8)$$

or alternatively:

$$\operatorname{ctn} \Delta\phi_1 = \frac{1 - B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)}{B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)} \qquad (10.6.3\text{-}9)$$

The (10.6.3-8) - (10.6.3-9) Ψ_1 maximum relations are then used in (10.6.3-3) for Ψ_1. First we expand the leading cosine function in the (10.6.3-3) Ψ_1 expression as:

$$\begin{aligned} & \cos\left(\Delta\phi_1 + \phi_{\omega ARS_x} - \phi_{aAccl_y}\right) = \\ & \quad \cos \Delta\phi_1 \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) - \sin \Delta\phi_1 \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \end{aligned} \qquad (10.6.3\text{-}10)$$

With (10.6.3-10), Ψ_1 in (10.6.3-3) becomes:

$$\begin{aligned}\Psi_1 = &\left[B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) - 1\right] \cos \Delta\phi_1 \\ &- B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \sin \Delta\phi_1\end{aligned} \tag{10.6.3-11}$$

Substituting the (10.6.3-8) maximum relation in (10.6.3-11) yields:

$$\begin{aligned}\Psi_1 &= - B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \operatorname{ctn} \Delta\phi_1 \cos \Delta\phi_1 \\ &\quad - B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \sin \Delta\phi_1 \\ &= - B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) \left(\operatorname{ctn} \Delta\phi_1 \cos \Delta\phi_1 + \sin \Delta\phi_1\right)\end{aligned} \tag{10.6.3-12}$$

The $\Delta\phi_1$ trigonometric function in (10.6.3-12) can be condensed as:

$$\operatorname{ctn} \Delta\phi_1 \cos \Delta\phi_1 + \sin \Delta\phi_1 = \csc \Delta\phi_1 = \pm\sqrt{1 + \operatorname{ctn}^2\Delta\phi_1} \tag{10.6.3-13}$$

Substituting the (10.6.3-9) maximum Ψ_1 relation gives for $1 + \operatorname{ctn}^2\Delta\phi_1$ in (10.6.3-13):

$$\begin{aligned}1 + \operatorname{ctn}^2\Delta\phi_1 &= 1 + \left[\frac{1 - B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)}{B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)}\right]^2 \\ &= \frac{\left\{\begin{aligned}&\left[B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)\right]^2 \\ &+ \left[1 - B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)\right]^2\end{aligned}\right\}}{\left[B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)\right]^2} \\ &= \frac{\left\{\begin{aligned}&\left[B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)\right]^2 + 1 \\ &- 2 B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right) + \left[B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)\right]^2\end{aligned}\right\}}{\left[B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)\right]^2} \\ &= \frac{1 + B^2_{\omega ARS_x} B^2_{aAccl_y} - 2 B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)}{\left[B_{\omega ARS_x} B_{aAccl_y} \sin\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)\right]^2}\end{aligned} \tag{10.6.3-14}$$

We substitute (10.6.3-13) with (10.6.3-14) (using the negative sign for the square root) into (10.6.3-12) to find the maximum value for Ψ_1:

$$\Psi_1 = \sqrt{1 + B^2_{\omega ARS_x} B^2_{aAccl_y} - 2 B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)} \qquad (10.6.3\text{-}15)$$

The identical process for Ψ_2 in (10.6.3-3) (but selecting the positive sign for the square root) yields for the minimum:

$$\Psi_2 = - \sqrt{1 + B^2_{\omega ARS_y} B^2_{aAccl_x} - 2 B_{\omega ARS_y} B_{aAccl_x} \cos\left(\phi_{\omega ARS_y} - \phi_{aAccl_x}\right)} \qquad (10.6.3\text{-}16)$$

Finally, we substitute (10.6.3-15) - (10.6.3-16) into the (10.6.3-3) Ψ expression and use the result in (10.6.3-1) to obtain the worst case solution for $\delta\dot{v}_{SFScul/SnsDyn_Z}$:

$$\delta\dot{v}_{SF/Scul/SnsDyn_Z} = \qquad (10.6.3\text{-}17)$$

$$\frac{1}{2} p^2_{Vib_0} \left\{ B_{\theta_x} B_{a_{SF_y}} \sqrt{1 + B^2_{\omega ARS_x} B^2_{aAccl_y} - 2 B_{\omega ARS_x} B_{aAccl_y} \cos\left(\phi_{\omega ARS_x} - \phi_{aAccl_y}\right)} \right.$$
$$\left. + B_{\theta_y} B_{a_{SF_x}} \sqrt{1 + B^2_{\omega ARS_y} B^2_{aAccl_x} - 2 B_{\omega ARS_y} B_{aAccl_x} \cos\left(\phi_{\omega ARS_y} - \phi_{aAccl_x}\right)} \right\}$$

Equation (10.6.3-17) is now in a form that can be applied to our simulation program. As for the sculling term analysis presented earlier, we use the Figure 10.5.1-1 simplified model for the dynamic characteristics (as in Equation (10.6.1-2)) and approximate particular terms in (10.6.3-17) as:

$$B_{\theta_x} = B_{\vartheta}(\Omega) \qquad B_{a_{SF_y}} = B_A(\Omega) \qquad B_{a_{SF_x}} = 0 \qquad (10.6.3\text{-}18)$$

Then the $\delta\dot{v}_{SF/Scul/SnsDyn_Z}$ response to discrete sinusoidal acceleration inputs is from (10.6.3-18):

$$\delta\dot{v}_{SF/Scul/SnsDyn_Z} = \qquad (10.6.3\text{-}19)$$

$$\frac{1}{2} a^2_{Vib_0} B_{\vartheta}(\Omega) B_A(\Omega) \sqrt{\begin{pmatrix} 1 + B^2_{\omega ARS_x}(\Omega) B^2_{aAccl_y}(\Omega) \\ - 2 B_{\omega ARS_x}(\Omega) B_{aAccl_y}(\Omega) \cos\left(\phi_{\omega ARS_x}(\Omega) - \phi_{aAccl_y}(\Omega)\right) \end{pmatrix}}$$

in which the Ω frequency dependence of the dynamic response terms has now been explicitly identified. As in Section 10.6.1 we note that a similar result is obtained along axis X under the same Y axis vibration due to Z axis angular rate sensor dynamic response characteristics:

$$\delta\dot{v}_{SF/Scul/SnsDyn_x} = \tag{10.6.3-20}$$

$$\frac{1}{2}\, a_{Vib_0}^2\, B_\vartheta(\Omega)\, B_A(\Omega) \sqrt{\begin{pmatrix} 1 + B_{\omega ARS_z}^2(\Omega)\, B_{aAccl_y}^2(\Omega) \\ - 2\, B_{\omega ARS_z}(\Omega)\, B_{aAccl_y}(\Omega) \cos\left(\phi_{\omega ARS_z}(\Omega) - \phi_{aAccl_y}(\Omega)\right) \end{pmatrix}}$$

where

$\delta\dot{v}_{SF/Scul/SnsDyn_x}$ = B Frame X component of $\delta\underline{\dot{v}}_{SF/Scul_{SnsDyn}}$.

$B_{\omega ARS_z}(\Omega)$, $\phi_{\omega ARS_z}(\Omega)$ = Amplitude and phase dynamic frequency response characteristics of the Z axis angular rate sensor.

Values for $B_\vartheta(\Omega)$, $B_A(\Omega)$ in (10.6.3-19) would be provided from Equations (10.6.1-8). The $B_{\omega ARS_x}(\Omega)$, $B_{\omega ARS_z}(\Omega)$, $B_{aAccl_y}(\Omega)$, $\phi_{\omega ARS_x}(\Omega)$, $\phi_{\omega ARS_z}(\Omega)$, $\phi_{aAccl_y}(\Omega)$ inertial sensor dynamic response characteristics would be provided as a simulation program input (as a function of Ω) or from an analytical model deemed representative of the inertial sensor dynamic response. Equations (10.6.3-19) and (10.6.3-20) are now in the form that can be added to the (10.6.1-23) list of performance parameters computed by the simulation for discrete frequency vibration acceleration inputs.

The equivalent form of (10.6.3-19) - (10.6.3-20) for addition to our simulation program random vibration performance parameter list in Equations (10.6.1-25) is:

$$\mathcal{E}\left(\delta\dot{v}_{SF/Scul/SnsDyn_z}\right) =$$

$$\int_0^\infty B_\vartheta(\omega)\, B_A(\omega) \sqrt{\begin{pmatrix} 1 + B_{\omega ARS_x}^2(\omega)\, B_{aAccl_y}^2(\omega) \\ - 2\, B_{\omega ARS_x}(\omega)\, B_{aAccl_y}(\omega) \cos\left(\phi_{\omega ARS_x}(\omega) - \phi_{aAccl_y}(\omega)\right) \end{pmatrix}}\; G_{aVib}(\omega)\, d\omega$$

$$\tag{10.6.3-21}$$

$$\mathcal{E}\left(\delta\dot{v}_{SF/Scul/SnsDyn_x}\right) =$$

$$\int_0^\infty B_\vartheta(\omega)\, B_A(\omega) \sqrt{\begin{pmatrix} 1 + B_{\omega ARS_z}^2(\omega)\, B_{aAccl_y}^2(\omega) \\ - 2\, B_{\omega ARS_z}(\omega)\, B_{aAccl_y}(\omega) \cos\left(\phi_{\omega ARS_z}(\omega) - \phi_{aAccl_y}(\omega)\right) \end{pmatrix}}\; G_{aVib}(\omega)\, d\omega$$

11 Strapdown Algorithm Validation

11.0 OVERVIEW

A key aspect of the strapdown inertial navigation software design process is the validation of the computational subroutines. In general this consists of operating the subroutines in a test computer against simulated strapdown sensor inputs with a corresponding navigation parameter profile (e.g., attitude, velocity, position). The navigation parameter solution generated with the strapdown subroutines under test is compared numerically against the equivalent navigation profile parameters to validate the subroutines.

The success of the validation depends on the accuracy of the reference navigation solution profile accompanying the simulated sensor input data. Ideally, the reference solution should be exact (i. e., completely error free). In addition, the reference solution profile(s) should be designed to exercise all elements of the computational routines under test. In general, this dictates reference profile(s) that do not represent realistic conditions encountered in normal navigation system use. It also generally involves several simulation profiles, each designed to exercise different groupings of the computational routines under test.

Algorithm validation simulators can be grouped into two classes; general simulators that exercise groupings of algorithms based on characteristic motion that is independent of the algorithm design, and specialized simulators based on particular design characteristics of the algorithms to be validated. In some cases, a general simulator can also serve as a specialized simulator for a particular algorithm evaluation.

This chapter provides examples of specialized and general simulators for validating strapdown algorithms based on the author's direct experience. Table 11.2-1 in Section 11.2 lists the strapdown computational routines that can be validated with each general simulator discussed.

11.1 SPECIALIZED VALIDATION SIMULATIONS

Specialized validation simulations are typically developed by the design engineer for particular algorithm designs to assure that the developed algorithms are analytically correct and in compliance with budgeted accuracy requirements. In this section we will provide some examples of specialized validation techniques that might be used during the algorithm design process, each of which would be executed using a specialized simulation designed for the purpose.

For example, consider the (7.1.1.3-1) and (7.1.1.3-10) direction cosine matrix orthogonality/normality correction algorithm and the quaternion normality correction algorithm given by Equations (7.1.2.3-6) and (7.1.2.3-8). These algorithms can be verified by inputting a direction cosine matrix and attitude quaternion that contain intentional orthogonality and normality error. For the direction cosine matrix correction algorithm input, a non-orthogonal/non-normal direction cosine matrix $\widehat{C}$ can be calculated from a proper arbitrarily selected reference direction cosine matrix C (without orthogonality/normality error) using (3.5.1-1) with the error matrix E set to a small arbitrary symmetric matrix (See Section 3.5.1 E_{SYM} discussion for rationale). For the quaternion correction algorithm input, a non-normal quaternion $\widehat{q}$ can be obtained from a proper arbitrarily selected reference quaternion q by multiplying q by 1 plus an arbitrarily selected small constant. After applying (7.1.1.3-1), (7.1.1.3-10) and (7.1.2.3-6), (7.1.2.3-8) to $\widehat{C}$ and $\widehat{q}$, the result should be C and q (within the second order error approximations used in the correction algorithm development).

An effective method for uncovering algorithm error is to compare the output of the algorithm under test with that of another algorithm designed to perform the same basic function. For example, the Equation (7.1.1.1-3) direction cosine matrix algorithm can be compared against the equivalent (7.1.2.1-3) quaternion equivalent using (7.1.2.4-1) to convert the quaternion to the equivalent direction cosine form for the comparison. Alternatively, (3.2.4.3-9) with (3.2.4.3-1) can be used to convert the direction cosine matrix to the equivalent quaternion form for comparison. The comparison should result in an error that is compatible with the truncation error used in the algorithm power series expansion terms. The $\underline{\phi}_m$ rotation vector input to (7.1.1.1-3) and (7.1.2.1-3) for the test would be identical and set to some non-trivial arbitrary value. For safety, the test should be repeated with different $\underline{\phi}_m$ components and signs.

Another method of algorithm validation utilizes a reversibility routine to derive the algorithm input from its output. An exact comparison of the derived input compared with the actual input validates the algorithm under test and its inverse. For example, the output from the Equation (7.1.1.1-3) direction cosine matrix algorithm can be input to inversion Equations (3.2.2.2-17) - (3.2.2.2-19) to obtain the (7.1.1.1-3) input rotation vector $\underline{\phi}_m$. The $\underline{\phi}_m$ vector calculated by the inversion process should match the original input to (7.1.1.1-3) within the algorithm power series truncation error. The original $\underline{\phi}_m$ rotation vector input to (7.1.1.1-3) would be set to some non-trivial arbitrary value. For safety, the test should be repeated with different $\underline{\phi}_m$ components and signs.

Another example of the previous output inversion test method is validation of the Equation (4.1.2-1) direction cosine matrix to ϕ, θ, ψ Euler angle extraction algorithm by calculating the direction cosine matrix input to (4.1.2-1) with the (3.2.3.1-2) Euler angle to direction cosine matrix routine. The (4.1.2-1) ϕ, θ, ψ output should match the (3.2.3.1-2) ϕ, θ, ψ input for various arbitrarily selected values of ϕ, θ, ψ.

During the design of a new unfamiliar algorithm, it may be desirable that an unusual analytical expression be validated by simulation. For example, consider Equations (8.2.1.1-7) - (8.2.1.1-9) which form the basis for coning algorithm compensation of inertial sensor error. Equations (8.2.1.1-7) - (8.2.1.1-9) can be validated by simulation comparison with the Equation (8.2.1.1-6) equivalent, using arbitrarily defined values for the K_{Mis} angular rate sensor misalignment matrix. A similar process can be used to validate the equivalent but more complicated sculling algorithm sensor compensation terms in Section 8.2.2.1.

In some cases, the analytical form of an integration algorithm can be validated by comparison with an exact closed-form integral of the fundamental equation from which it was derived. For example, consider the (7.1.1.1-12) $\underline{\phi}_m$ rotation vector computation for attitude updating using input from B Frame integrated angular rate and coning algorithms (7.1.1.1.1-17) - (7.1.1.1.1-18). Equations (7.1.1.1.1-17) - (7.1.1.1.1-18) were derived directly from Equation (7.1.1.1-13) which, when combined with (7.1.1.1-12) is:

$$\underline{\phi}_m = \int_{t_{m-1}}^{t_m} \left[\underline{\omega}_{IB}^{B} + \frac{1}{2} \left(\underline{\alpha}(t) \times \underline{\omega}_{IB}^{B} \right) \right] dt \qquad (11.1\text{-}1)$$

$$\underline{\alpha}(t) = \int_{t_{m-1}}^{t} \underline{\omega}_{IB}^{B} \, d\tau \qquad (11.1\text{-}2)$$

The Equation (7.1.1.1.1-17) - (7.1.1.1.1-18) algorithms were derived to be exact under general linearly ramping angular rate:

$$\underline{\omega}_{IB}^{B} = \underline{A} + \underline{B}\,(t - t_{m-1}) \qquad (11.1\text{-}3)$$

where

$\underline{A}, \underline{B}$ = Constants.

Equation (11.1-1) can be analytically integrated using (11.1-2) and (11.1-3) for input to yield:

$$\underline{\phi}_m = \underline{A}\,(t_m - t_{m-1}) + \frac{1}{2}\,\underline{B}\,(t_m - t_{m-1})^2 + \frac{1}{12}\left(\underline{A} \times \underline{B}\right)(t_m - t_{m-1})^3 \qquad (11.1\text{-}4)$$

Equation (11.1-4) can now be used as a truth model for comparison with the algorithm solution for $\underline{\phi}_m$ computed from (7.1.1.1-12), (7.1.1.1.1-17) and (7.1.1.1.1-18). The result should be an identical match for arbitrary values of $\underline{A}$ and $\underline{B}$. The input $\Delta\underline{\alpha}_l$ to the algorithm for this test would be from (11.1-3) and (7.1.1.1.1-17):

$$\Delta\underline{\alpha}_l = \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}^{B}\, dt = \underline{A}\,(t_l - t_{l-1}) + \frac{1}{2}\,\underline{B}\left((t_l - t_{m-1})^2 - (t_{l-1} - t_{m-1})^2\right) \tag{11.1-5}$$

A similar procedure can be used to validate the analytical integrity of B Frame $\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$ and $\Delta\underline{R}_{SF_m}^{B}$ integration algorithms for velocity and position updating. The $\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$ algorithms derive from Equation (7.2.2.2-5):

$$\Delta\underline{v}_{SF_m}^{B_{I(m-1)}} = \int_{t_{m-1}}^{t_m} \left[I + \left(\underline{\alpha}(t)\times\right)\right] \underline{a}_{SF}^{B}\, dt \tag{11.1-6}$$

with (11.1-2) for $\underline{\alpha}(t)$. The $\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$ algorithms presented in Section 7.2 were designed to be analytically exact under general linearly ramping angular rate (e.g., as in (11.1-3)) and specific force:

$$\underline{a}_{SF}^{B} = \underline{C} + \underline{D}\,(t - t_{m-1}) \tag{11.1-7}$$

where

$\underline{C}, \underline{D}$ = Constants.

Analytically performing the (11.1-6) integration with input from (11.1-2), (11.1-3) and (11.1-7) yields for a $\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$ truth model:

$$\begin{aligned}\Delta\underline{v}_{SF_m}^{B_{I(m-1)}} &= \underline{C}\,(t_m - t_{m-1}) + \frac{1}{2}\left(\underline{D} + \underline{A}\times\underline{C}\right)(t_m - t_{m-1})^2 \\ &\quad + \frac{1}{3}\left[\underline{A}\times\underline{D} + \frac{1}{2}\left(\underline{B}\times\underline{C}\right)\right](t_m - t_{m-1})^3 + \frac{1}{8}\left(\underline{B}\times\underline{D}\right)(t_m - t_{m-1})^4\end{aligned} \tag{11.1-8}$$

Two equivalent algorithm versions of $\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$ have been developed in Section 7.2 based on the identically same general linearly ramping angular-rate/specific-force conditions; Equations (7.1.1.1.1-17), (7.2.2.2-23), (7.2.2.2-25), (7.2.2.2.2-14), (7.2.2.2.2-15); and Equations (7.1.1.1.1-17), (7.2.2.2-26), (7.2.2.2.2-23), (7.2.2.2.2-24). Inputs to these algorithms are $\Delta\underline{\alpha}_l$

and $\Delta\underline{\upsilon}_l$ which, for the validation test, would be calculated with (11.1-5) and from (7.2.2.2.2-14) with (11.1-7):

$$\Delta\underline{\upsilon}_l = \int_{t_{l-1}}^{t_l} \underline{a}_{SF}^{B} dt = \underline{C}\,(t_l - t_{l-1}) + \frac{1}{2}\,\underline{D}\left((t_l - t_{m-1})^2 - (t_{l-1} - t_{m-1})^2\right) \tag{11.1-9}$$

The $\Delta\underline{v}_{SF_m}^{B_{I(m-1)}}$ algorithm solutions using (11.1-5) and (11.1-9) input should identically match the Equation (11.1-8) result for arbitrary values of $\underline{A}$, $\underline{B}$, $\underline{C}$, and $\underline{D}$.

The $\Delta\underline{R}_{SF_m}^{B}$ algorithm derives from Equations (7.3.3-4) and (7.2.2.2-5):

$$\Delta\underline{R}_{SF_m}^{B} = \int_{t_{m-1}}^{t_m} \Delta\underline{v}_{SF}^{B(m-1)}(t)\, dt \tag{11.1-10}$$

$$\Delta\underline{v}_{SF}^{B(m-1)}(t) = \int_{t_{m-1}}^{t} \left[I + \left(\underline{\alpha}(\tau)\times\right)\right] \underline{a}_{SF}^{B} d\tau \tag{11.1-11}$$

with (11.1-2) for $\underline{\alpha}(t)$. Analytically performing the (11.1-10) integration with input from (11.1-2), (11.1-3), (11.1-7) and (11.1-11) yields for the $\Delta\underline{R}_{SF_m}^{B}$ truth model:

$$\begin{aligned} \Delta\underline{R}_{SF_m}^{B} = {} & \frac{1}{2}\,\underline{C}\,(t_m - t_{m-1})^2 + \frac{1}{6}\left(\underline{D} + \underline{A}\times\underline{C}\right)(t_m - t_{m-1})^3 \\ & + \frac{1}{12}\left[\underline{A}\times\underline{D} + \frac{1}{2}\left(\underline{B}\times\underline{C}\right)\right](t_m - t_{m-1})^4 + \frac{1}{40}\left(\underline{B}\times\underline{D}\right)(t_m - t_{m-1})^5 \end{aligned} \tag{11.1-12}$$

The algorithm versions for $\Delta\underline{R}_{SF_m}^{B}$ developed in Section 7.3 are also based on general linearly ramping angular-rate/specific-force conditions, and are provided by Equations (7.1.1.1.1-17), (7.2.2.2.2-14), (7.2.2.2.2-15), (7.3.3-9), (7.3.3-11), and (7.3.3.2-18) - (7.3.3.2-20). Inputs to these algorithms are $\Delta\underline{\alpha}_l$ and $\Delta\underline{\upsilon}_l$ which, for the validation test, would be calculated with (11.1-5) and (11.1-9). The $\Delta\underline{R}_{SF_m}^{B}$ algorithm solution using (11.1-5) and (11.1-9) input should identically match the Equation (11.1-12) result for arbitrary values of $\underline{A}$, $\underline{B}$, $\underline{C}$, and $\underline{D}$.

As a final example, closed-form solutions can also be found similarly for the velocity/position translation vectors ($\underline{\eta}$ and $\underline{\zeta}$) of Chapter 19, Section 19.1 based on continuous form algorithm set c differential Equations (19.1.8-3). Assuming linearly ramping angular rate and specific force as in (11.1-3) and (11.1-7), the integral of Equations (19.1.8-3) gives (in addition to (11.1-4) for the rotation vector):

$$\underline{\eta}_m = \underline{C}\,(t_m - t_{m-1}) + \frac{1}{2}\,\underline{D}\,(t_m - t_{m-1})^2 + \frac{1}{12}\left(\underline{A} \times \underline{D} - \underline{B} \times \underline{C}\right)(t_m - t_{m-1})^3$$

$$\begin{aligned} \underline{\zeta}_m = {} & \frac{1}{2}\,\underline{C}\,(t_m - t_{m-1})^2 + \frac{1}{6}\,\underline{D}\,(t_m - t_{m-1})^3 \\ & + \frac{1}{72}\left(2\,\underline{A} \times \underline{D} - 3\,\underline{B} \times \underline{C}\right)(t_m - t_{m-1})^4 - \frac{1}{360}\,\underline{B} \times \underline{D}\,(t_m - t_{m-1})^5 \end{aligned} \tag{11.1-13}$$

Digital algorithms for calculating $\underline{\eta}$ and $\underline{\zeta}$ provided in Equations (19.1.11-1) are also based on general linearly ramping angular-rate/specific-force. Inputs to these algorithms are $\Delta\underline{\alpha}_l$ and $\Delta\underline{\upsilon}_l$ which, for the validation test, would be calculated with (11.1-5) and (11.1-9). The $\underline{\eta}$, $\underline{\zeta}$ algorithm solution using (11.1-5) and (11.1-9) input should identically match the Equation (11.1-13) result for arbitrary values of $\underline{A}$, $\underline{B}$, $\underline{C}$, and $\underline{D}$.

11.2 GENERAL STRAPDOWN ALGORITHM VALIDATION SIMULATORS

This section describes four general strapdown reference simulation routines for validating particular groupings of strapdown inertial navigation computational routines. The simulator routines are based on the author’s direct experience. Each simulator has the common characteristic of having closed-form exact solution outputs that are analytical functions of the time since simulation start. As such, their outputs are void of potential error produced by digital integration (in contrast with the strapdown algorithms they are used to validate, which are inherently digital integration routines by nature). The simulators to be described are denoted as:

SPIN-CONE - Simulates combined sensor frame spinning/coning angular motion. Used to test the attitude integration routines associated with strapdown angular rate sensor inputs under generalized dynamic spinning/coning conditions.

SPIN-ACCEL - Simulates constant angular rate spinning of the sensor and attitude reference frame axes under constant sensor frame specific force acceleration conditions. Used to test the attitude integration and acceleration transformation routines for constant strapdown angular rate sensor and accelerometer inputs (i.e., without coning and sculling).

SPIN-ROCK-SIZE - Simulates generalized spinning/rocking of the sensor frame from an offset lever arm about a fixed rotation axis to create generalized spinning/rocking/acceleration motion. Includes lever arms to three arbitrary sensor frame locations to simulate accelerometer location and associated size effect. Used to test the attitude integration, accelerometer size effect correction, acceleration transformation and specific force acceleration integration/double-integration routines for spinning/oscillating/accelerating sensor frame angular rates and accelerations about a generalized non-rotating axis (i.e., with zero coning but with non-zero sculling and scrolling).

GEN NAV - Simulates generalized motion over the surface of an ellipsoidal shaped earth including acceleration to a specified velocity, climb maneuver to a specified altitude, long term cruise at the specified altitude with specified oscillatory altitude changes, and including a segment of constant (selectable) sensor frame angular rate about an arbitrary (selectable) axis. Used to test the overall integrated strapdown inertial navigation routines under generalized motion (exclusive of coning/sculling/scrolling motion).

The validation test computer program that uses the above simulators would be structured to call the strapdown routines being tested in the sequence and cycle rate they would normally be processed in the actual INS navigation computer (e.g., Table 7.5-1). Each of the above simulators would be programmed into the validation test program as an additional computational subroutine called from the point in the strapdown routine call sequence that input inertial sensor data is received. When called, the simulator subroutine would calculate and deliver simulated angular rate sensor and accelerometer data to the strapdown routines under test, in addition to the exact navigation parameters associated with the sensor data at that instant of time. Comparison routines would then be utilized at strategically located points in the strapdown routine call sequence to assess the accuracy of the strapdown routine generated navigation parameters compared with the exact navigation parameter data.

Table 11.2-1 lists basic computational functions implemented in strapdown inertial navigation software packages together with a list of representative computational routines (from Tables 7.5-1 and 8.4-1) utilized for their implementation. Table 11.2-1 identifies (with a check ✔) the Table 7.5-1 strapdown navigation routines and Table 8.4-1 accelerometer size effect compensation routines that can be validated with the simulators. The sub-sections following Table 11.2-1 describe each of the simulators and the method used to compare their output parameters against comparable data obtained from the strapdown routines under test.

Table 11.2-1 Strapdown Inertial Navigation System Computational Algorithms Validated With Indicated Simulators

ALGORITHM FUNCTION	EQUATION	SIMULATOR USED FOR VALIDATION			
		SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
HIGH SPEED CALCULATIONS					
Integrated B Frame Angular Rate Increments	(7.1.1.1.1-17)	✔		✔	✔
Integrated B Frame Acceleration Increments	(7.2.2.2.2-14)			✔	✔

(Continued)

ALGORITHM FUNCTION	EQUATION	SIMULATOR USED FOR VALIDATION: SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
Coning Increment	(7.1.1.1.1-18)	✔			
Sculling Increment	(7.2.2.2.2-15)			✔	
Doubly Integrated B Frame Angular Rate And Acceleration Increments (For High Resolution Position Algorithm)	(7.3.3.2-18), (7.3.3.2-19)			✔	✔
Scrolling Increment (For High Resolution Position Algorithm)	(7.3.3.2-20)			✔	
Accelerometer Size Effect Input Parameters	(8.1.4.1.4-5)			✔ (With $\Delta\underline{\alpha}'_l = \Delta\underline{\alpha}_l$)	

NORMAL SPEED CALCULATIONS

INERTIAL SENSOR COMPENSATION

ALGORITHM FUNCTION	EQUATION	SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
Accelerometer Size Effect Compensation Term Components	(8.1.4.1.1.1-15)			✔	
Accelerometer Size Effect Compensation Terms	(8.1.4.1.1-11), (8.1.4.1.2-13)			✔ (With $\Delta\delta\underline{\upsilon}_{SizeC_m} = 0$)	
Integrated And Doubly Integrated B Frame Acceleration Increments Compensated For Accelerometer Size Effect	(8.1.2.2-4), (8.1.2.2-6)			✔	

EARTH RELATED PARAMETERS

ALGORITHM FUNCTION	EQUATION	SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
Earth Polar Axis Component Of Geodetic Vertical Unit Vector	(5.3-16)				✔
Modified Radial Distance To Earth Surface Location	(5.1-10)				✔
Radial Distance To Earth Surface Location	(5.2.1-4)				✔
Radial Distance To Navigation Point	(5.2.1-5)				✔

ALGORITHM FUNCTION	EQUATION	SIMULATOR USED FOR VALIDATION			
		SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
Cosine Of Range Vector Polar Coordinate Angle	(5.2.2-3)				✔
Modified Sine Of Range Vector Polar Coordinate Angle	(5.2.2-5)				✔
Cosine And Modified Sine Of Difference Between Geocentric And Geodetic Latitudes	(5.2.3-5)				✔
Local Earth Surface Point Radius Of Curvature In Latitude Direction	(5.2.4-25)				✔
Local Navigation Point Radius Of Curvature In Latitude Direction	(5.2.4-37)				✔
Curvature Matrix In The N Frame	(5.3-18)				✔
Vertical Transport Rate Component	Section 4.5 For Options				✔
N Frame Transport Rate Vector	(5.3-17)				✔
Gravity Components In Polar Coordinates	(5.4-1) & (5.4-2)				✔
North And Vertical Gravity Components	(5.4-4)				✔
North And Vertical Plumb-bob Gravity Components	(5.4.1-9)				✔
N Frame Plumb-bob Gravity Components	(5.4.1-11)				✔
N Frame Earth Rate Vector	(4.1.1-3), (4.1.1-4)				✔
<u>VELOCITY CALCULATIONS</u>					
B Frame Velocity Rotation Compensation (Exact Formulation)	(7.2.2.2.1-7), (7.2.2.2.1-8)		✔	✔	✔
B Frame Velocity Rotation Compensation (First Order Approximation Form)	(7.2.2.2-25)		✔	✔	✔

ALGORITHM FUNCTION	EQUATION	SIMULATOR USED FOR VALIDATION			
		SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
B Frame Integrated Specific Force Acceleration Increment	(8.1.4.1-14)		✔	✔	✔
L Frame Integrated Specific Force Acceleration Increment	(7.2.2-2)		✔	✔	✔
L Frame Rotation Vector	(7.2.2.1-4) - (7.2.2.1-7)				✔
L Frame Rotation Matrix (First Order Form)	(7.2.2.1-1)		✔		✔
L Frame Rotation Compensation	(7.2.2-4)		✔		✔
Integrated Coriolis Acceleration & Plumb-bob Gravity Increment	(7.2.1-1) - (7.2.1-3)				✔
N Frame Velocity Update	(7.2-2)			✔ With Zero For Coriolis & Gravity	✔
Vertical Channel Control Signals	(7.2-6)				✔
Vertical Velocity Divergence Control	(7.2-5)				✔
East, North, Up Velocity Component Outputs	(4.3.1-4)				✔
<u>POSITION CALCULATIONS</u>					
Position Rotation Compensation (High Resolution Position Algorithm - Exact Form)	(7.3.3.1-16)			✔	✔
Position Rotation Compensation (High Resolution Position Algorithm - First Order Accuracy Form)	(7.3.3-11)			✔	✔
Body Frame Position Increment Due To Specific Force Acceleration (High Resolution Position Algorithm)	(8.1.4.1-16)			✔	✔
N Frame Position Increment (High Resolution Position Algorithm)	(7.3.3-8)			✔	✔

ALGORITHM FUNCTION	**EQUATION**	**SIMULATOR USED FOR VALIDATION**			
		SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
N Frame Position Increment (Trapezoidal Position Algorithm)	(7.3.2-1)			✔	✔
Altitude Change	(7.3.1.-3)				✔
Position Rotation Vector	(7.3.1-11), (7.3.1-12)				✔
Position Rotation Change Matrix	(7.3.1-8)				✔
Altitude Update	(7.3.1-1)				✔
Altitude Divergence Control	(7.3.1-5)				✔
Position Direction Cosine Matrix Update	(7.3.1-6)				✔
Latitude, Longitude Outputs And Wander Angle	(4.4.2.1-3)				✔
<u>ATTITUDE CALCULATIONS</u>					
B Frame Rotation vector	(7.1.1.1-12)	✔	✔	✔	✔
B Frame Rotation Matrix (For Attitude Direction Cosine Matrix Updating)	(7.1.1.1-3)	✔	✔	✔	✔
B Frame Rotation Quaternion (For Attitude Quaternion Updating)	(7.1.2.1-3)	✔	✔	✔	✔
Attitude Update For B Frame Rotation (Direction Cosine Matrix Form)	(7.1.1.1-1)	✔	✔	✔	✔
Attitude Update For B Frame Rotation (Quaternion Form)	(7.1.2.1-1)	✔	✔	✔	✔
L Frame Rotation Vector	(7.1.1.2.1-3), (7.1.1.2.1-5)				✔
L Frame Rotation Matrix For Attitude Direction Cosine Matrix Updating (Exact Form)	(7.1.1.2-3)		✔		✔
L Frame Quaternion For Attitude Quaternion Updating (Exact Form)	(7.1.2.2-3)		✔		✔

ALGORITHM FUNCTION	EQUATION	SIMULATOR USED FOR VALIDATION			
		SPIN-CONE	SPIN-ACCEL	SPIN-ROCK-SIZE	GEN NAV
Attitude Update For L Frame Rotation (Direction Cosine Matrix Form)	(7.1.1.2-1)		✔		✔
Attitude Update For L Frame Rotation (Quaternion Form)	(7.1.2.2-1)		✔		✔
Normalization And Orthogonalization Corrections (For Attitude Direction Cosine Matrix)	(7.1.1.3-1), (7.1.1.3-10)	✔	✔	✔	✔
Normalization Corrections (For Attitude Quaternion)	(7.1.2.3-6), (7.1.2.3-8)	✔	✔	✔	✔
Attitude Quaternion To Attitude Direction Cosine Matrix Conversion (For Attitude Quaternion As Basic Attitude Form)	(7.1.2.4-1)	✔	✔	✔	✔
Roll, Pitch, True Heading Euler Angle Outputs	(4.1.2-1), (4.1.2-2)	✔	✔	✔	✔

11.2.1 SPIN-CONE SIMULATOR

The Spin-Cone simulator provides exact closed-form attitude and corresponding continuous integrated body frame angular rates for a spinning body with coning motion. The difference between integrated body rates at successive strapdown software sensor sampling cycles simulate the inputs from strapdown angular rate sensors used in the attitude update routines for the software under test. The Spin-Cone attitude direction cosine matrix is compared with the strapdown software computed direction cosine matrix to establish strapdown software algorithm attitude accuracy in terms of normality, orthogonality, and misalignment errors.

11.2.1.1 ANALYTICAL MODEL

The Spin-Cone simulator is based on a closed-form solution to the attitude motion described by a body spinning at a fixed magnitude rotation rate and whose spin axis is rotating at a fixed precessional rate. The geometry of the motion is described in Figure 11.2.1.1-1.

Figure 11.2.1.1-1 shows the spin-axis and precessional-axis to be separated by an angle β. The spin axis rotates about the precessional axis which is defined to be perpendicular to a non-rotating inertial plane. A set of body reference axes is implied in Figure 11.2.1.1-1 that rotates relative to a defined set of non-rotating coordinates where:

L = Non-rotating coordinate frame that is fixed to the non-rotating plane with X_L, Y_L axes in the plane and the Z_L axis perpendicular to the plane in the direction opposite to the precessional rate vector.

R = Body "reference" coordinate axes fixed to the body with the X axis (X_R) along the spin axis. The R Frame is at a fixed orientation relative to the traditional sensor assembly B Frame axes defined in Section 2.2. A distinction is made between the B and R Frames so that the angular rate generated by the Figure 11.2.1.1-1 motion can have selected projections on the B Frame sensor axes to test the general response of the strapdown attitude algorithms.

β = Angle between the precessional axis and the R-Frame X_R spin axis (the "cone angle") - considered constant.

ω_s = Inertial rotation rate of the body about X_R ("spin rate") - considered constant.

ω_c = Inertial precessional rate of the body X_R axis about the precessional axis.

ϕ, θ, ψ = Roll, pitch, heading Euler angles of the body R Frame axes relative to the L Frame (See Section 3.2.3 for the definition of Euler angles).

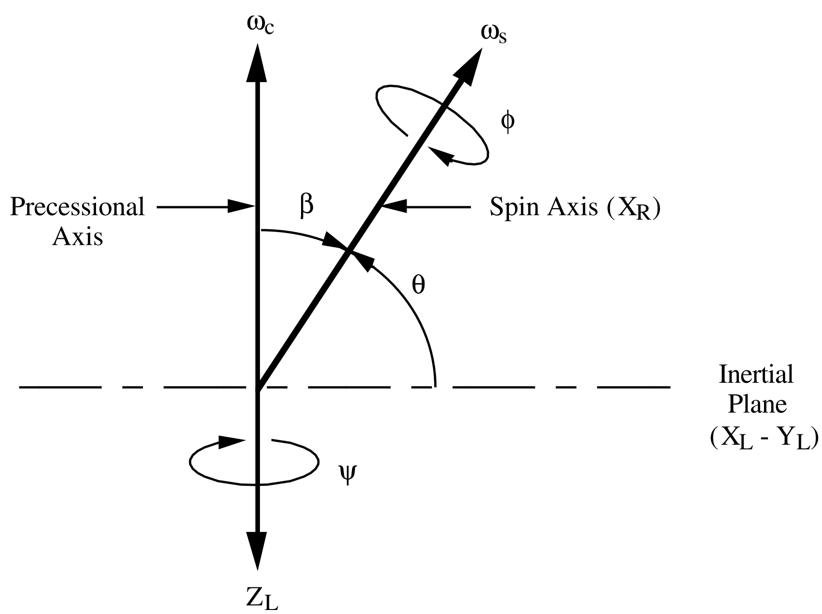


Figure 11.2.1.1-1 Spin-Cone Geometry

Note from the Figure 11.2.1.1-1 geometry, that:

$$\theta = \pi / 2 - \beta = \text{Constant} \tag{11.2.1.1-1}$$

The total inertial rotation rate of the body is the vector sum of the Euler angle rotation rate effects. Generalized Equations (3.3.3.1-3) (with the R Frame in this section corresponding to the B Frame and the A Frame rates set to zero as being inertially fixed) show that (with $\dot{\theta} = 0$ from Equation (11.2.1.1-1)) the R Frame inertial rotation rate components are given by:

$$\omega_{XR} = \dot{\phi} - \dot{\psi} \sin \theta \qquad \omega_{YR} = \dot{\psi} \cos \theta \sin \phi \qquad \omega_{ZR} = \dot{\psi} \cos \theta \cos \phi \tag{11.2.1.1-2}$$

where

ω_{iR} = R Frame i-axis component of inertial body rotation rate.

From the geometry in Figure 11.2.1.1-1 we also note that:

$$\omega_{XR} = \omega_s \qquad \dot{\psi} = - \omega_c \tag{11.2.1.1-3}$$

Combining (11.2.1.1-1) - (11.2.1.1-3) yields:

$$\begin{aligned} \dot{\phi} &= \omega_s - \omega_c \cos \beta & \omega_{XR} &= \omega_s \\ \dot{\theta} &= 0 & \omega_{YR} &= - \omega_c \sin \beta \sin \phi \\ \dot{\psi} &= - \omega_c & \omega_{ZR} &= - \omega_c \sin \beta \cos \phi \end{aligned} \tag{11.2.1.1-4}$$

The integral of (11.2.1.1-4) provides the formulas for describing the body B Frame attitude history and the integrated body rates that will form the basis for calculating simulated strapdown angular rate sensor output signals. Recognizing from (11.2.1.1-4) and the definitions of ω_s and ω_c, that the Euler angle rates are each constant, yields for the integral:

$$\begin{aligned} \phi &= \left(\omega_s - \omega_c \cos \beta\right) t + \alpha & I\omega_{XR} &= \omega_s t \\ \theta &= \pi / 2 - \beta & I\omega_{YR} &= \omega_c \sin \beta \frac{1}{\dot{\phi}} \left(\cos \phi - \cos \alpha\right) \\ \psi &= - \omega_c t & I\omega_{ZR} &= - \omega_c \sin \beta \frac{1}{\dot{\phi}} \left(\sin \phi - \sin \alpha\right) \end{aligned} \tag{11.2.1.1-5}$$

where

α = Initial value for ϕ. The initial value for ψ is assumed to be zero.

$I_{\omega iR}$ = Integral of ω_{iR} from simulation start.

t = Time from simulation start.

The integral of the body R Frame rates over successive sample cycles is then computed as the difference between the current and past values of $I_{\omega iR}$:

$$\Delta I\omega_{iR_l} = I\omega_{iR_l} - I\omega_{iR_{l-1}} \tag{11.2.1.1-6}$$

where

$\Delta I_{\omega iR}$ = Integrated R Frame rotation rate component i over successive angular rate sensor triad sample cycles (i.e., integrated rate increment).

l = Angular rate sensor triad sample time index.

The attitude of the R Frame relative to the L Frame corresponding with the Equation (11.2.1.1-6) angular rate sensor signals is defined by the ϕ, θ, ψ Euler angles in Equations (11.2.1.1-5), or from the equivalent direction cosine matrix using generalized Equations (3.2.3.1-2):

$$\begin{aligned}
C_{RL_{11}} &= \cos\theta \cos\psi \\
C_{RL_{12}} &= -\cos\phi \sin\psi + \sin\phi \sin\theta \cos\psi \\
C_{RL_{13}} &= \sin\phi \sin\psi + \cos\phi \sin\theta \cos\psi \\
\\
C_{RL_{21}} &= \cos\theta \sin\psi \\
C_{RL_{22}} &= \cos\phi \cos\psi + \sin\phi \sin\theta \sin\psi \\
C_{RL_{23}} &= -\sin\phi \cos\psi + \cos\phi \sin\theta \sin\psi \\
\\
C_{RL_{31}} &= -\sin\theta \\
C_{RL_{32}} &= \sin\phi \cos\theta \\
C_{RL_{33}} &= \cos\phi \cos\theta
\end{aligned} \tag{11.2.1.1-7}$$

where

$C_{RL_{ij}}$ = Element in row i column j of the C_R^L matrix defined below.

C_R^L = Direction cosine matrix that transforms vectors from the R Frame to the L Frame.

The ω_s, ω_c, β, α terms in Equations (11.2.1.1-5) are Spin-Cone simulator input parameters.

11.2.1.2 SIMULATED STRAPDOWN ANGULAR RATE SENSOR OUTPUTS

The strapdown angular rate sensor axes (B Frame) are assumed to be oriented relative to the body reference axes (R Frame) by a fixed set of Euler angles, where:

$J_{\psi R}, J_{\theta R}, J_{\phi R}$ = Fixed Euler angles about Z, Y, X R Frame axes describing the body angular rate sensor B Frame orientation relative to the body reference R Frame.

The associated direction cosine matrix describing the B Frame attitude relative to the R Frame is as defined by generalized Equations (3.2.3.1-2):

$$\begin{aligned}
C_{BR_{11}} &= \cos J_{\theta R} \cos J_{\psi R} \\
C_{BR_{12}} &= -\cos J_{\phi R} \sin J_{\psi R} + \sin J_{\phi R} \sin J_{\theta R} \cos J_{\psi R} \\
C_{BR_{13}} &= \sin J_{\phi R} \sin J_{\psi R} + \cos J_{\phi R} \sin J_{\theta R} \cos J_{\psi R} \\[4pt]
C_{BR_{21}} &= \cos J_{\theta R} \sin J_{\psi R} \\
C_{BR_{22}} &= \cos J_{\phi R} \cos J_{\psi R} + \sin J_{\phi R} \sin J_{\theta R} \sin J_{\psi R} \\
C_{BR_{23}} &= -\sin J_{\phi R} \cos J_{\psi R} + \cos J_{\phi R} \sin J_{\theta R} \sin J_{\psi R} \\[4pt]
C_{BR_{31}} &= -\sin J_{\theta R} \\
C_{BR_{32}} &= \sin J_{\phi R} \cos J_{\theta R} \\
C_{BR_{33}} &= \cos J_{\phi R} \cos J_{\theta R}
\end{aligned} \tag{11.2.1.2-1}$$

where

$C_{BR_{ij}}$ = Element in row i column j of C_B^R defined below.

C_B^R = Direction cosine matrix that transforms vectors from the B Frame to the R Frame.

The body angular rate sensor axis (B Frame) components of body rotation rate are related to the body reference axis R Frame components through:

$$\underline{\omega}^B = C_R^B \, \underline{\omega}^R \tag{11.2.1.2-2}$$

where

$\underline{\omega}^B$ = B Frame (angular rate sensor axis) components of angular rotation rate.

$\underline{\omega}^R$ = Body reference R Frame components of angular rotation rate (X, Y, Z components given by $\omega_{XR}, \omega_{YR}, \omega_{ZR}$).

Because C_R^B is constant, the integral of (11.2.1.2-2) over an inertial sensor data sample cycle is:

$$\Delta I\underline{\omega}^B = C_R^B \, \Delta I\underline{\omega}^R = \left(C_B^R\right)^T \Delta I\underline{\omega}^R \tag{11.2.1.2-3}$$

where

$\Delta I \underline{\omega}^R$ = Integrated R Frame rates between angular rate sensor triad sample cycles (components i = X, Y, Z given by $\Delta I \omega_{iR}$ in Equation (11.2.1.1-6)).

$\Delta I \underline{\omega}^B$ = Integrated B Frame angular rates between angular rate sensor triad sample cycles.

The $\Delta I \underline{\omega}^B$ components from (11.2.1.2-3) represent the integrated B Frame angular rate increments that simulate strapdown angular rate sensor outputs over a sensor sample cycle. These signals are used as inputs to the strapdown attitude integration algorithms under test (e.g., representing the $\Delta \underline{\alpha}_l$ vector in Equations (7.1.1.1.1-17)).

11.2.1.3 ATTITUDE DIRECTION COSINE AND EULER ANGLE OUTPUTS

The simulated strapdown angular rate sensor axis B Frame direction cosine matrix relative to the L Frame is given from the Equation (3.2.1-5) chain rule by:

$$C_B^L = C_R^L \, C_B^R \tag{11.2.1.3-1}$$

where

C_B^L = Direction cosine matrix that transforms vectors from the B Frame to the L Frame.

The components of C_R^L are defined by Equations (11.2.1.1-7) and the components of C_B^R are defined by Equations (11.2.1.2-1).

The Euler angles associated with C_B^L are evaluated by application of generalized Equations (3.2.3.2-1), (3.2.3.2-2) and (3.2.3.2-4):

$$\theta_{BL} = \tan^{-1} \frac{- C_{31}}{\sqrt{C_{32}^2 + C_{33}^2}}$$

For $\left| C_{31} \right| < 0.999$:

$$\phi_{BL} = \tan^{-1} \frac{C_{32}}{C_{33}} \qquad \psi_{BL} = \tan^{-1} \frac{C_{21}}{C_{11}}$$

For $C_{31} \leq - 0.999$: (11.2.1.3-2)

$$\psi_{BL} - \phi_{BL} = \tan^{-1} \frac{C_{23} - C_{12}}{C_{13} + C_{22}}$$

For $C_{31} \geq 0.999$:

$$\psi_{BL} + \phi_{BL} = \pi + \tan^{-1} \frac{C_{23} + C_{12}}{C_{13} - C_{22}}$$

where

C_{ij} = Element in row i and column j of C_B^L.

ϕ_{BL}, θ_{BL}, ψ_{BL} = Roll, pitch, heading Euler angles associated with C_B^L.

11.2.1.4 STRAPDOWN ATTITUDE ALGORITHM ERROR EVALUATION

The error in the attitude data generated from the strapdown software algorithms under test can be evaluated by comparison of the ϕ, θ, ψ software generated Euler angle output parameters with the equivalent Euler angle data provided from the reference simulator (Spin-Cone or other). Additionally, the attitude direction cosine matrix (DCM) calculated by the algorithms under test can be compared with equivalent data from the reference simulator. This section describes a computational routine that can be used to compare the DCM computed with the strapdown inertial navigation algorithms under test against the reference DCM provided by the Spin-Cone (or other) simulator. The comparison calculates normalization, orthogonality and misalignment errors of the strapdown algorithm DCM.

The DCM comparison routine equations are based on generalized Equations (3.5.1-6) of Section 3.5.1 which state that the error in a direction cosine matrix can be characterized by:

$$
\begin{aligned}
&C_{ALG} = (I + E)\, C_{REF} \\
&E = C_{ALG}\, C_{REF}^T - I \\
&E = E_{SYM} + E_{SKSYM} \\
&E_{SYM} = \frac{1}{2}\left(E + E^T\right) \qquad E_{SKSYM} = \frac{1}{2}\left(E - E^T\right)
\end{aligned}
\tag{11.2.1.4-1}
$$

where

C_{ALG} = Direction cosine matrix calculated with the strapdown algorithms under test.

C_{REF} = Direction cosine matrix provided from the reference simulator (i.e., Spin-Cone or other).

E = Matrix containing C_{ALG} errors.

E_{SYM} = Symmetrical portion of E.

E_{SKSYM} = Skew symmetrical portion of E.

I = Identity matrix.

Section 3.5.1 shows that the diagonal elements of E_{SYM} equal the normality errors in the rows of C_{ALG} and that the off-diagonal elements equal half the orthogonality errors between the rows of C_{ALG} (See Equations (3.5.1-11), (3.5.1-12) and (3.5.1-14)). Section 3.5.1 also shows that E_{SKSYM} (to first order) is the skew symmetric form of the orientation error vector associated with the misalignment of the C_{ALG} matrix from C_{REF}.

Based on the previous understanding, we can now identify particular elements of E_{SYM} and E_{SKSYM} as normalization, orthogonality and misalignment errors associated with the C_{ALG} matrix:

$$
\begin{aligned}
&\text{Row 1 Normalization Error} = E_{SYM_{11}} \\
&\text{Row 2 Normalization Error} = E_{SYM_{22}} \\
&\text{Row 3 Normalization Error} = E_{SYM_{33}} \\
&\\
&\text{Orthogonality Error Between Rows 1 And 2} = 2\,E_{SYM_{12}} = 2\,E_{SYM_{21}} \\
&\text{Orthogonality Error Between Rows 1 And 3} = 2\,E_{SYM_{13}} = 2\,E_{SYM_{31}} \\
&\text{Orthogonality Error Between Rows 2 And 3} = 2\,E_{SYM_{23}} = 2\,E_{SYM_{32}} \\
&\\
&\text{Misalignment Error Around L Frame axis X} = E_{SKSYM_{32}} = -\,E_{SKSYM_{23}} \\
&\text{Misalignment Error Around L Frame axis Y} = E_{SKSYM_{13}} = -\,E_{SKSYM_{31}} \\
&\text{Misalignment Error Around L Frame axis Z} = E_{SKSYM_{21}} = -\,E_{SKSYM_{12}}
\end{aligned}
\qquad (11.2.1.4\text{-}2)
$$

where

$E_{SYM_{ij}}$, $E_{SKSYM_{ij}}$ = Elements in row i and column j of E_{SYM}, E_{SKSYM}.

11.2.2 SPIN-ACCEL SIMULATOR

The Spin-Accel simulator provides exact closed-form attitude, velocity and corresponding body frame integrated angular rate and acceleration increments for a sensor assembly undergoing a constant body frame linear-acceleration/angular-rate maneuver. The body frame integrated angular rate and acceleration increments simulate inputs from strapdown angular rate sensors and accelerometers used in the attitude-update/acceleration-transformation routines for the software under test. The Spin-Accel direction cosine matrix represents the true body (B Frame) attitude relative to an attitude reference L Frame. The Spin-Accel attitude direction cosine matrix is compared with the strapdown software computed direction cosine matrix to establish strapdown software attitude accuracy in terms of normality, orthogonality and misalignment errors (as in Section 11.2.1.4). To verify acceleration transformation/integration operations, the Spin-Accel velocity vector provided in the L Frame is compared against the integrated transformed acceleration (B Frame to L Frame) obtained using the strapdown software routines. The Spin-Accel simulator also provides for a user specified constant rotation rate of the L Frame.

11.2.2.1 ANALYTICAL MODEL

The basic analytical function performed by the Spin-Accel simulator is to provide a closed-form solution for the integral of transformed constant B Frame acceleration:

$$\underline{v}^{L}_{SF} \;=\; \int_0^t C^{L}_{B}\, \underline{a}^{B}_{SF}\, d\tau \;=\; \left(\int_0^t C^{L}_{B}\, d\tau \right) \underline{a}^{B}_{SF} \qquad (11.2.2.1\text{-}1)$$

where

t = Current time.

τ = Integration running time parameter.

B = Body (or strapdown inertial sensor) coordinate frame at time τ.

L = Attitude reference coordinate frame at time τ.

C^{L}_{B} = Instantaneous direction cosine matrix at time τ that transforms vectors from body to attitude reference coordinates.

$\underline{a}^{B}_{SF}$ = Body frame specific force acceleration (constant).

$\underline{v}^{L}_{SF}$ = Integrated specific force acceleration in the L Frame from simulation start time $t = 0$ to the current time t.

The C^{L}_{B} matrix in (11.2.2.1-1) can be decomposed as follows using the Equation (3.2.1-5) chain rule:

$$C^{L}_{B} \;=\; C^{L}_{L_0}\, C^{L_0}_{B_0}\, C^{B_0}_{B} \qquad (11.2.2.1\text{-}2)$$

where

B_0 = B Frame orientation in non-rotating inertial space at simulation time $t = 0$.

L_0 = L Frame orientation in non-rotating inertial space at simulation time $t = 0$.

The $C^{B_0}_{B}$ and $C^{L}_{L_0}$ matrices in (11.2.2.1-2) can be expanded in terms of rotation vectors by application of Equations (3.2.2-1) and (3.2.2.1-4) - (3.2.2.1-7):

$$\begin{aligned} C^{B_0}_{B} \;&=\; I + \sin\phi \left(\underline{u}^{B}_{\phi}\times\right) + (1 - \cos\phi)\left(\underline{u}^{B}_{\phi}\times\right)\left(\underline{u}^{B}_{\phi}\times\right) \\ C^{L}_{L_0} \;&=\; \left(C^{L_0}_{L}\right)^{T} \;=\; I - \sin\zeta \left(\underline{u}^{L}_{\zeta}\times\right) + (1 - \cos\zeta)\left(\underline{u}^{L}_{\zeta}\times\right)\left(\underline{u}^{L}_{\zeta}\times\right) \end{aligned} \qquad (11.2.2.1\text{-}3)$$

with

$$\begin{aligned} \underline{\phi} \;&=\; \phi\, \underline{u}^{B}_{\phi} \qquad & \phi \;&=\; \sqrt{\underline{\phi}\cdot\underline{\phi}} \qquad & \underline{u}^{B}_{\phi} \;&=\; \underline{\phi} / \phi \\ \underline{\zeta} \;&=\; \zeta\, \underline{u}^{L}_{\zeta} \qquad & \zeta \;&=\; \sqrt{\underline{\zeta}\cdot\underline{\zeta}} \qquad & \underline{u}^{L}_{\zeta} \;&=\; \underline{\zeta} / \zeta \end{aligned} \qquad (11.2.2.1\text{-}4)$$

where

$\underline{\phi}$ = Rotation vector describing the orientation of Frame B relative to Frame B_0.

$\underline{u}_{\phi}^{B}$ = Unit vector along the $\underline{\phi}$ rotation vector.

ϕ = Magnitude of $\underline{\phi}$.

$\underline{\zeta}$ = Rotation vector describing the orientation of Frame L relative to Frame L_0.

$\underline{u}_{\zeta}^{L}$ = Unit vector along the $\underline{\zeta}$ rotation vector.

ζ = Magnitude of $\underline{\zeta}$.

For the Spin-Accel simulator, the angular rates of the B Frame and L Frame relative to non-rotating inertial space are constant. The integral solution from simulation time $t = 0$ to generalized rotation vector rate Equation (3.3.5-14) under constant angular rate conditions (as can be verified by substitution - See last paragraph in Section 7.1.1.1 for further explanation) is:

$$\underline{\phi} = \underline{\omega}_{\phi}\,\tau \qquad\qquad \underline{\zeta} = \underline{\omega}_{\zeta}\,\tau \tag{11.2.2.1-5}$$

where

$\underline{\omega}_{\phi}$ = B Frame angular rate relative to non-rotating inertial space (constant).

$\underline{\omega}_{\zeta}$ = L Frame angular rate relative to non-rotating inertial space (constant).

Combining Equations (11.2.2.1-5) and (11.2.2.1-4) then provides:

$$\begin{aligned} \omega_{\phi} &= \sqrt{\underline{\omega}_{\phi} \cdot \underline{\omega}_{\phi}} \qquad & \phi &= \omega_{\phi}\,\tau \qquad & \underline{u}_{\phi}^{B} &= \underline{\omega}_{\phi} / \omega_{\phi} \\ \omega_{\zeta} &= \sqrt{\underline{\omega}_{\zeta} \cdot \underline{\omega}_{\zeta}} \qquad & \zeta &= \omega_{\zeta}\,\tau \qquad & \underline{u}_{\zeta}^{L} &= \underline{\omega}_{\zeta} / \omega_{\zeta} \end{aligned} \tag{11.2.2.1-6}$$

where

ω_{ϕ} = Magnitude of $\underline{\omega}_{\phi}$ (constant).

ω_{ζ} = Magnitude of $\underline{\omega}_{\zeta}$ (constant).

Equations (11.2.2.1-6) also show that the $\underline{u}_{\phi}^{B}$, $\underline{u}_{\zeta}^{L}$ vectors represent unit vectors along $\underline{\omega}_{\phi}$ and $\underline{\omega}_{\zeta}$, which are constant because $\underline{\omega}_{\phi}$ and $\underline{\omega}_{\zeta}$ are constant.

We now substitute the ϕ, ζ expressions from (11.2.2.1-6) into (11.2.2.1-3) to obtain:

$$C_{B}^{B_0} = I + \Delta C_{B}^{B_0} \qquad\qquad C_{L_0}^{L} = I + \Delta C_{L_0}^{L} \tag{11.2.2.1-7}$$

in which

$$\begin{aligned}
\Delta C_{B}^{B_0} &= \sin\omega_{\phi}\tau\left(\underline{u}_{\phi}^{B}\times\right)+\left(1-\cos\omega_{\phi}\tau\right)\left(\underline{u}_{\phi}^{B}\times\right)\left(\underline{u}_{\phi}^{B}\times\right) \\
\Delta C_{L_0}^{L} &= -\sin\omega_{\zeta}\tau\left(\underline{u}_{\zeta}^{L}\times\right)+\left(1-\cos\omega_{\zeta}\tau\right)\left(\underline{u}_{\zeta}^{L}\times\right)\left(\underline{u}_{\zeta}^{L}\times\right)
\end{aligned} \tag{11.2.2.1-8}$$

With Equations (11.2.2.1-7) and (11.2.2.1-2), the C_B^L integral in Equation (11.2.2.1-1) becomes:

$$\begin{aligned}
\int_0^t C_B^L\, d\tau &= C_{B_0}^{L_0}\, t + C_{B_0}^{L_0}\int_0^t \Delta C_B^{B_0}\, d\tau + \left(\int_0^t \Delta C_{L_0}^{L}\, d\tau\right) C_{B_0}^{L_0} \\
&\quad + \int_0^t \Delta C_{L_0}^{L}\, C_{B_0}^{L_0}\, \Delta C_B^{B_0}\, d\tau
\end{aligned} \tag{11.2.2.1-9}$$

or

$$\int_0^t C_B^L\, d\tau = C_{B_0}^{L_0}\, t + \Delta C_{\phi} + \Delta C_{\zeta} + \Delta C_{\zeta\phi} \tag{11.2.2.1-10}$$

with

$$\begin{aligned}
\Delta C_{\phi} &\equiv C_{B_0}^{L_0}\int_0^t \Delta C_B^{B_0}\, d\tau \qquad\qquad \Delta C_{\zeta} \equiv \left(\int_0^t \Delta C_{L_0}^{L}\, d\tau\right) C_{B_0}^{L_0} \\
\Delta C_{\zeta\phi} &\equiv \int_0^t \Delta C_{L_0}^{L}\, C_{B_0}^{L_0}\, \Delta C_B^{B_0}\, d\tau
\end{aligned} \tag{11.2.2.1-11}$$

In performing the integrations in (11.2.2.1-11) with (11.2.2.1-8) for $\Delta C_B^{B_0}$ and $\Delta C_{L_0}^{L}$, it is convenient to first define the following integral functional elements:

$$\begin{aligned}
f_1(\omega t) &= \int_0^t \sin\omega\tau\, d\tau = \frac{1}{\omega}\left(1-\cos\omega t\right) \\
f_2(\omega t) &= \int_0^t \left(1-\cos\omega\tau\right) d\tau = t - \frac{1}{\omega}\sin\omega t
\end{aligned} \tag{11.2.2.1-12}$$

where

$f_1(\omega t)$, $f_2(\omega t)$ = Generalized integral functional elements.

ω = Generalized angular rate argument.

Applying Equations (11.2.2.1-12) to ΔC_{ϕ} and ΔC_{ζ} in (11.2.2.1-11) with (11.2.2.1-8) then yields:

$$
\begin{aligned}
\Delta C_{\phi} &= C_{B_0}^{L_0}\left[f_1(\omega_\phi t)\left(\underline{u}_\phi^B\times\right) + f_2(\omega_\phi t)\left(\underline{u}_\phi^B\times\right)\left(\underline{u}_\phi^B\times\right)\right] \\
\Delta C_{\zeta} &= \left[- f_1(\omega_\zeta t)\left(\underline{u}_\zeta^L\times\right) + f_2(\omega_\zeta t)\left(\underline{u}_\zeta^L\times\right)\left(\underline{u}_\zeta^L\times\right)\right] C_{B_0}^{L_0}
\end{aligned}
\tag{11.2.2.1-13}
$$

Evaluation of the $\Delta C_{\zeta\phi}$ term in (11.2.2.1-11) is more involved and makes use of the following trigonometric identities:

$$
\begin{aligned}
\sin(\alpha + \beta) &= \sin\alpha\cos\beta + \cos\alpha\sin\beta \\
\cos(\alpha + \beta) &= \cos\alpha\cos\beta - \sin\alpha\sin\beta
\end{aligned}
\tag{11.2.2.1-14}
$$

Proceeding from $\Delta C_{\zeta\phi}$ in (11.2.2.1-11) with (11.2.2.1-8) and (11.2.2.1-14) then obtains:

$$
\begin{aligned}
\Delta C_{\zeta\phi} = &\int_0^t \left[- \sin\omega_\zeta\tau\left(\underline{u}_\zeta^L\times\right) + (1 - \cos\omega_\zeta\tau)\left(\underline{u}_\zeta^L\times\right)\left(\underline{u}_\zeta^L\times\right)\right] C_{B_0}^{L_0}\left[\sin\omega_\phi\tau\left(\underline{u}_\phi^B\times\right)\right. \\
&\qquad\qquad \left. + (1 - \cos\omega_\phi\tau)\left(\underline{u}_\phi^B\times\right)\left(\underline{u}_\phi^B\times\right)\right] d\tau \\
= &- \left(\int_0^t \sin\omega_\zeta\tau\,\sin\omega_\phi\tau\,d\tau\right)\left(\underline{u}_\zeta^L\times\right) C_{B_0}^{L_0}\left(\underline{u}_\phi^B\times\right) \\
&+ \left(\int_0^t (1 - \cos\omega_\zeta\tau)\sin\omega_\phi\tau\,d\tau\right)\left(\underline{u}_\zeta^L\times\right)\left(\underline{u}_\zeta^L\times\right) C_{B_0}^{L_0}\left(\underline{u}_\phi^B\times\right) \\
&- \left(\int_0^t \sin\omega_\zeta\tau\,(1 - \cos\omega_\phi\tau)\,d\tau\right)\left(\underline{u}_\zeta^L\times\right) C_{B_0}^{L_0}\left(\underline{u}_\phi^B\times\right)\left(\underline{u}_\phi^B\times\right) \\
&+ \left(\int_0^t (1 - \cos\omega_\zeta\tau)(1 - \cos\omega_\phi\tau)\,d\tau\right)\left(\underline{u}_\zeta^L\times\right)\left(\underline{u}_\zeta^L\times\right) C_{B_0}^{L_0}\left(\underline{u}_\phi^B\times\right)\left(\underline{u}_\phi^B\times\right) \\
= &- \left(\int_0^t \frac{1}{2}\left[\cos(\omega_\zeta - \omega_\phi)\tau - \cos(\omega_\zeta + \omega_\phi)\tau\right] d\tau\right)\left(\underline{u}_\zeta^L\times\right) C_{B_0}^{L_0}\left(\underline{u}_\phi^B\times\right) \\
&+ \left(\int_0^t \left(\sin\omega_\phi\tau - \frac{1}{2}\left[\sin(\omega_\zeta + \omega_\phi)\tau - \sin(\omega_\zeta - \omega_\phi)\tau\right]\right) d\tau\right)\left(\underline{u}_\zeta^L\times\right)^2 C_{B_0}^{L_0}\left(\underline{u}_\phi^B\times\right)
\end{aligned}
\tag{11.2.2.1-15}
$$

(Continued)

$$- \left(\int_0^t \left(\sin \omega\zeta \tau - \frac{1}{2} \left[\sin\,(\omega\zeta + \omega_\phi)\tau + \sin\,(\omega\zeta - \omega_\phi)\tau \right] \right) d\tau \right) \left(\underline{u}_\zeta^L \times \right) C_{B_0}^{L_0} \left(\underline{u}_\phi^B \times \right) \left(\underline{u}_\phi^B \times \right) \qquad \begin{matrix} (11.2.2.1\text{-}15) \\ (\text{Continued}) \end{matrix}$$

$$+ \left(\int_0^t \left(1 + \frac{1}{2} \left[\cos\,(\omega\zeta - \omega_\phi)\tau + \cos\,(\omega\zeta + \omega_\phi)\tau \right] - \cos \omega\zeta \tau - \cos \omega_\phi \tau \right) d\tau \right) \left(\underline{u}_\zeta^L \times \right)^2 C_{B_0}^{L_0} \left(\underline{u}_\phi^B \times \right)^2$$

With (11.2.2.1-12), Equation (11.2.2.1-15) then becomes:

$$\begin{aligned} \Delta C_{\zeta\phi} = {} & \frac{1}{2} \left[f_2\big((\omega\zeta - \omega_\phi)t\big) - f_2\big((\omega\zeta + \omega_\phi)t\big) \right] \left(\underline{u}_\zeta^L \times \right) C_{B_0}^{L_0} \left(\underline{u}_\phi^B \times \right) \\ & + \frac{1}{2} \left[2\, f_1(\omega_\phi t) - f_1\big((\omega\zeta + \omega_\phi)t\big) + f_1\big((\omega\zeta - \omega_\phi)t\big) \right] \left(\underline{u}_\zeta^L \times \right)^2 C_{B_0}^{L_0} \left(\underline{u}_\phi^B \times \right) \qquad (11.2.2.1\text{-}16) \\ & - \frac{1}{2} \left[2\, f_1(\omega\zeta t) - f_1\big((\omega\zeta + \omega_\phi)t\big) - f_1\big((\omega\zeta - \omega_\phi)t\big) \right] \left(\underline{u}_\zeta^L \times \right) C_{B_0}^{L_0} \left(\underline{u}_\phi^B \times \right) \left(\underline{u}_\phi^B \times \right) \\ & - \frac{1}{2} \left[f_2\big((\omega\zeta - \omega_\phi)t\big) + f_2\big((\omega\zeta + \omega_\phi)t\big) - 2\, f_2(\omega\zeta t) - 2\, f_2(\omega_\phi t) \right] \left(\underline{u}_\zeta^L \times \right)^2 C_{B_0}^{L_0} \left(\underline{u}_\phi^B \times \right)^2 \end{aligned}$$

Finally, we combine Equations (11.2.2.1-1) and (11.2.2.1-10) to obtain the closed-form output expression for $\underline{v}_{SF}^L$:

$$\underline{v}_{SF}^L = \left(C_{B_0}^{L_0}\, t + \Delta C_\phi + \Delta C_\zeta + \Delta C_{\zeta\phi} \right) \underline{a}_{SF}^B \qquad (11.2.2.1\text{-}17)$$

with $C_{B_0}^{L_0}$ provided by user specification and $\Delta C_\phi, \Delta C_\zeta, \Delta C_{\zeta\phi}$ provided by Equations (11.2.2.1-13) and (11.2.2.1-16). Inputs to $\Delta C_\phi, \Delta C_\zeta, \Delta C_{\zeta\phi}$ Equations (11.2.2.1-13) and (11.2.2.1-16) are the user specified B Frame and L Frame angular rate vectors ($\underline{\omega}_\phi$, $\underline{\omega}\zeta$), conversion Equations (11.2.2.1-6) for ϕ, $\underline{u}_\phi^B$, ζ, $\underline{u}_\zeta^B$, user specified initial direction cosine matrix $C_{B_0}^{L_0}$ and the time since simulation start t.

The Spin-Accel simulator C_B^L attitude direction cosine matrix at time t corresponding to $\underline{v}_{SF}^L$ in Equation (11.2.2.1-17) is given by Equation (11.2.2.1-2) using the $C_{B_0}^{L_0}$ matrix specified by the user and $C_B^{B_0}$, $C_{L_0}^L$ as provided by Equations (11.2.2.1-3) and (11.2.2.1-5) (with $\tau = t$). The Euler angles associated with C_B^L are evaluated as in Equations (11.2.1.3-2).

User specification of the $C_{B_0}^{L_0}$ matrix is easily accomplished by definition of three Euler angles describing the B Frame attitude relative to the L Frame at time t = 0 and then applying generalized Equations (3.2.3.1-2):

$$
\begin{aligned}
C_{11_0} &= \cos\theta_0 \cos\psi_0 \\
C_{12_0} &= -\cos\phi_0 \sin\psi_0 + \sin\phi_0 \sin\theta_0 \cos\psi_0 \\
C_{13_0} &= \sin\phi_0 \sin\psi_0 + \cos\phi_0 \sin\theta_0 \cos\psi_0 \\
\\
C_{21_0} &= \cos\theta_0 \sin\psi_0 \\
C_{22_0} &= \cos\phi_0 \cos\psi_0 + \sin\phi_0 \sin\theta_0 \sin\psi_0 \\
C_{23_0} &= -\sin\phi_0 \cos\psi_0 + \cos\phi_0 \sin\theta_0 \sin\psi_0 \\
\\
C_{31_0} &= -\sin\theta_0 \\
C_{32_0} &= \sin\phi_0 \cos\theta_0 \\
C_{33_0} &= \cos\phi_0 \cos\theta_0
\end{aligned}
\tag{11.2.2.1-18}
$$

where

C_{ij_0} = Element in row i, column j of $C_{B_0}^{L_0}$.

ϕ_0, θ_0, ψ_0 = Roll, pitch, heading Euler angles associated with $C_{B_0}^{L_0}$.

11.2.2.2 SIMULATED STRAPDOWN INERTIAL SENSOR OUTPUTS

The Spin-Accel simulator must also provide B Frame integrated angular rate and specific force acceleration increments to the strapdown algorithms under test at the algorithm iteration cycle times to simulate the inputs from the strapdown angular rate sensors and accelerometers. Since the angular rate and specific force acceleration are constant, these signals are given simply by:

$$
\underline{\alpha} = \underline{\omega}_\phi T \qquad\qquad \underline{\upsilon} = \underline{a}_{SF}^{B} T \tag{11.2.2.2-1}
$$

where

$\underline{\alpha}$ = Simulated strapdown sensor assembly integrated angular rate sensor increment vector.

$\underline{\upsilon}$ = Simulated strapdown sensor assembly integrated accelerometer increment vector.

T = Strapdown algorithm attitude-update/acceleration-transformation time interval.

The $\underline{\alpha}$ and $\underline{\upsilon}$ vectors would then be used to represent the strapdown sensor inputs to the strapdown attitude-integration/acceleration-transformation algorithms under test (i.e., for the Chapter 7 and 19 (Section 19.1) algorithms, $\underline{\alpha}_m$ in Equation (7.1.1.1-12) with the coning term $\underline{\beta}_m$ set to zero, $\underline{\upsilon}_m$ in Equation (7.2.2.2-23) with the sculling term $\Delta\underline{v}_{Scul_m}$ set to zero, $\underline{\alpha}_m$, $\underline{\upsilon}_m$ for the rotation compensation term $\Delta\underline{v}_{Rot_m}$ as defined by Equation (7.2.2.2-25) or Equation (7.2.2.2.1-7), $\underline{\alpha}_m$ for $\underline{\phi}_m$ in Equations (19.1.5-9) (the equivalent to setting the coning term $\underline{\beta}_m$ to zero as discussed previously), and $\underline{\upsilon}_m$ in the Equation (19.1.11-1) $\underline{\eta}$ velocity translation vector expression with the $\delta\underline{\eta}_{Scul}$ sculling term set to zero for application in Equations (19.1.5-9)).

11.2.2.3 ATTITUDE REFERENCE FRAME ROTATION RATE OUTPUT

When evaluating the strapdown attitude-integration/acceleration-transformation algorithms with the Spin-Accel simulator, the rotation rate for the algorithm attitude reference frame must be set to match the user specified Spin-Accel L Frame rotation rate $\underline{\omega}_\zeta$ (e.g., setting $\underline{\omega}_{IL}^{L}$ to $\underline{\omega}_\zeta$ in Equations (7.1.1.2-4) and (7.2.2.1-2)).

11.2.2.4 STRAPDOWN ALGORITHM ERROR EVALUATION

The accuracy of the strapdown software attitude-integration/acceleration-transformation algorithms is evaluated with the Spin-Accel simulator by comparing the Spin-Accel attitude matrix C_B^L (and Euler angles) and the integrated L Frame specific force acceleration $\underline{v}_{SF}^{L}$ with the equivalent strapdown algorithm generated parameters at selected time points t.

The Spin-Accel simulator C_B^L matrix and associated Euler angles at time t (from Equations (11.2.2.1-2), (11.2.2.1-3), (11.2.2.1-5) and (11.2.2.1-18)) are compared with the C_B^L matrix and Euler angles calculated using the strapdown software attitude integration routines to assess the strapdown algorithm attitude accuracy. Section 11.2.1.4 describes how the comparison is made to evaluate strapdown algorithm C_B^L error in terms of normality, orthogonality and misalignment error parameters.

The Spin-Accel simulator integrated L Frame specific force acceleration $\underline{v}_{SF}^{L}$ at time t provided by Equation (11.2.2.1-17) is compared against the equivalent parameter calculated with the strapdown software algorithms under test as the output from the software attitude-integration/acceleration-transformation operations. For example, the L Frame integrated specific force acceleration increment $\Delta \underline{v}^{L}_{SF_m}$ calculated with the strapdown algorithms as in Equation (7.2.2-4) would be summed at the m cycle transformation rate to obtain the equivalent to $\underline{v}^{L}_{SF}$ for Spin-Accel comparison. The Chapter 19 Equations (19.1.3-2) unified approach $\underline{v}^{N}_{m}$ velocity algorithm (with $\Delta \underline{v}^{N}_{g_m}$ set to zero and the velocity translation vector $\underline{\eta}_m$ provided by (19.1.11-1)) can also be validated with this simulator by comparing $C^{L}_{N}\, \underline{v}^{N}_{m}$ to $\Delta \underline{v}^{L}_{SF_m}$.

11.2.3 SPIN-ROCK-SIZE SIMULATOR

The Spin-Rock-Size simulator provides exact closed form integrated angular rates, integrated linear accelerations, attitude, velocity and position simulating a strapdown sensor assembly undergoing spinning/sculling/scrolling dynamic motion with the individual accelerometers mounted at specified lever arm locations within the sensor assembly (i.e., simulating size effect separation). The integrated rates and accelerations are used as inputs to strapdown software algorithms under test to compute body attitude, accelerometer size effect lever arm compensation to the body navigation reference center, transformation of compensated specific force acceleration to navigation coordinates, and specific force acceleration integration to velocity and position. The Spin-Rock-Size simulator evaluates the strapdown software algorithm accuracy by comparing simulator computed position, velocity and attitude with the equivalent data generated by the strapdown software algorithms under test.

11.2.3.1 ANALYTICAL MODEL

The Spin-Rock-Size simulator generates navigation and inertial sensor outputs under dynamic motion around an arbitrarily specified and fixed rotation axis. The rotation axis is defined to be non-rotating and non-accelerating. The dynamic motion is characterized as rigid body motion around the specified axis with the specified axis located within the rotating rigid body. The strapdown sensor assembly being simulated is located in the rigid body and has its navigation reference center at a specified lever arm location from the rotation axis. Each accelerometer within the sensor assembly is located at an arbitrarily selected lever arm position. All accelerations measured by the accelerometers are created by the centripetal and tangential acceleration effects produced by their lever arm displacement from the rotation axis under rigid body dynamic angular motion around the rotation axis.

Based on the previous description, we define a fixed reference point located on the rotation axis and a lever arm fixed to the rigid body, emanating from the reference point, and terminating at an arbitrary point within the rigid body. The lever arm components can be analytically defined in inertial and body fixed coordinates as follows:

$$\underline{l}^{I} = C_B^I \, \underline{l}^{B} \tag{11.2.3.1-1}$$

where

$\underline{l}$ = Lever arm from the reference point to an arbitrary point in the rigid body.

I = Inertial non-rotating coordinate frame.

B = Body frame defined to be aligned with the inertial sensor assembly axes.

C_B^I = Direction cosine matrix that transforms vectors from the B Frame to the I Frame.

Taking the first derivative of Equation (11.2.3.1-1) yields:

$$\dot{\underline{l}}^{I} = \dot{C}_B^I \, \underline{l}^{B} + C_B^I \, \dot{\underline{l}}^{B} \tag{11.2.3.1-2}$$

Because $\underline{l}$ is defined as fixed in the B Frame, its rate of change in the B Frame is zero, hence, Equation (11.2.3.1-2) simplifies to:

$$\dot{\underline{l}}^{I} = \dot{C}_B^I \, \underline{l}^{B} \tag{11.2.3.1-3}$$

Generalized Equation (3.3.2-9) shows that:

$$\dot{C}_B^I = C_B^I \left(\underline{\omega}_{IB}^{B} \times \right) \tag{11.2.3.1-4}$$

where

$\underline{\omega}_{IB}^{B}$ = Angular rate of the B Frame relative to inertial space projected on B Frame axes.

so that (11.2.3.1-3) becomes:

$$\dot{\underline{l}}^{I} = C_B^I \left(\underline{\omega}_{IB}^{B} \times \right) \underline{l}^{B} \tag{11.2.3.1-5}$$

Taking the derivative of (11.2.3.1-5) and substituting (11.2.3.1-4) obtains:

$$\begin{aligned} \ddot{\underline{l}}^{I} &= \dot{C}_B^I \left(\underline{\omega}_{IB}^{B} \times \right) \underline{l}^{B} + C_B^I \left(\dot{\underline{\omega}}_{IB}^{B} \times \right) \underline{l}^{B} = C_B^I \left(\underline{\omega}_{IB}^{B} \times \right) \left(\underline{\omega}_{IB}^{B} \times \right) \underline{l}^{B} + C_B^I \left(\dot{\underline{\omega}}_{IB}^{B} \times \right) \underline{l}^{B} \\ &= C_B^I \left[\left(\dot{\underline{\omega}}_{IB}^{B} \times \right) + \left(\underline{\omega}_{IB}^{B} \times \right) \left(\underline{\omega}_{IB}^{B} \times \right) \right] \underline{l}^{B} \end{aligned} \tag{11.2.3.1-6}$$

Because the $\underline{l}$ emanation point has been defined to be inertially non-accelerating, the second derivative of $\underline{l}$ in the I Frame can be identified as the specific force acceleration at the end point of the $\underline{l}$ vector:

$$\underline{a}_{SF}^{I} \equiv \ddot{\underline{l}}^{I} \tag{11.2.3.1-7}$$

where

$\underline{a}_{SF}$ = Specific force acceleration of the end point of $\underline{l}$ (i.e., the acceleration that would be measured by an accelerometer located at the $\underline{l}$ end point).

Substituting (11.2.3.1-7) into (11.2.3.1-6) and transforming the result to the B Frame then shows that:

$$\underline{a}_{SF}^{B} = \left[\left(\dot{\underline{\omega}}_{IB}^{B}\times\right) + \left(\underline{\omega}_{IB}^{B}\times\right)\left(\underline{\omega}_{IB}^{B}\times\right)\right]\underline{l}^{B} \tag{11.2.3.1-8}$$

Equation (11.2.3.1-8) can now be specialized to a particular accelerometer location within the rigid body mounted strapdown sensor assembly. This is achieved by identifying a particular location within the sensor assembly as the "navigation center" for sensor assembly velocity/position determination with accelerometer locations defined relative to the navigation center:

$$\underline{a}_{SF_i}^{B} = \left[\left(\dot{\underline{\omega}}_{IB}^{B}\times\right) + \left(\underline{\omega}_{IB}^{B}\times\right)\left(\underline{\omega}_{IB}^{B}\times\right)\right]\left(\underline{l}_{0}^{B} + \underline{l}_{i}^{B}\right) \tag{11.2.3.1-9}$$

where

$\underline{l}_{0}^{B}$ = Displacement vector of the sensor assembly navigation center from the fixed reference point on the rotation axis.

$\underline{l}_{i}^{B}$ = Displacement vector of accelerometer i (i = 1, 2 or 3) from the sensor assembly navigation center.

$\underline{a}_{SF_i}^{B}$ = Specific force acceleration vector at the accelerometer i location.

We defined the rotation axis as itself non-rotating, hence, can write:

$$\underline{\omega}_{IB}^{B} = \dot{\gamma}\,\underline{u}_{\gamma}^{B} \qquad \dot{\underline{\omega}}_{IB}^{B} = \ddot{\gamma}\,\underline{u}_{\gamma}^{B} \tag{11.2.3.1-10}$$

where

$\underline{u}_{\gamma}^{B}$ = Unit vector along the rotation axis.

γ = Angular displacement of the sensor assembly around the rotation axis.

For the Spin-Rock-Size simulator, the γ angle is specified analytically as:

$$\gamma = A\,t + B\sin\Omega t \tag{11.2.3.1-11}$$

where

A, B, Ω = User selected constants.

11.2.3.2 SIMULATED STRAPDOWN INERTIAL SENSOR OUTPUTS

The simulated accelerometer i output is the component of $\underline{a}_{SF_i}^B$ along the accelerometer i input axis direction:

$$a_{SF_i} = \underline{u}_i^B \cdot \underline{a}_{SF_i}^B \qquad (11.2.3.2\text{-}1)$$

where

$\underline{u}_i^B$ = Unit vector along the accelerometer i input axis.

a_{SF_i} = Accelerometer i input.

Combining Equations (11.2.3.2-1), (11.2.3.1-9) and (11.2.3.1-10) yields:

$$a_{SF_i} = \underline{u}_i^B \cdot \left\{ \left[\ddot{\gamma} \left(\underline{u}_\gamma^B \times \right) + \dot{\gamma}^2 \left(\underline{u}_\gamma^B \times \right)^2 \right] \left(\underline{l}_0^B + \underline{l}_i^B \right) \right\} \qquad (11.2.3.2\text{-}2)$$

The output from accelerometer i is the integral of Equation (11.2.3.2-2) over a sensor assembly sample period:

$$\Delta\upsilon_{i_l} = \underline{u}_i^B \cdot \left\{ \left[F_{1_l} \left(\underline{u}_\gamma^B \times \right) + F_{2_l} \left(\underline{u}_\gamma^B \times \right)^2 \right] \left(\underline{l}_0^B + \underline{l}_i^B \right) \right\} \qquad (11.2.3.2\text{-}3)$$

with

$$\begin{aligned} F_{1_l} &\equiv \int_{t_{l-1}}^{t_l} \ddot{\gamma}\, dt = f_1(t_l) - f_1(t_{l-1}) \qquad & f_1(t) &\equiv \int \ddot{\gamma}\, dt = \dot{\gamma} \\ F_{2_l} &\equiv \int_{t_{l-1}}^{t_l} \dot{\gamma}^2\, dt = f_2(t_l) - f_2(t_{l-1}) \qquad & f_2(t) &\equiv \int \dot{\gamma}^2\, dt \end{aligned} \qquad (11.2.3.2\text{-}4)$$

where

$\Delta\upsilon_{i_l}$ = Accelerometer i output over the l^{th} sensor assembly sample period.

t_l, t_{l-1} = Time at the start and end of the l^{th} sensor assembly sample period.

The $f_1(t)$, $f_2(t)$ integral functions in (11.2.3.2-4) are evaluated from the analytical expression for γ in Equation (11.2.3.1-11).

The rate of change of γ is obtained as the time derivative of (11.2.3.1-11):

$$\dot{\gamma} = A + B\,\Omega \cos \Omega t \qquad (11.2.3.2\text{-}5)$$

Since $f_1(t)$ is used in (11.2.3.2-4) as the difference between its value at t_l and t_{l-1}, the A term in (11.2.3.2-5) cancels in the differencing operation, and f_1 in (11.2.3.2-4) can be redefined for simplicity as only containing the B term in (11.2.3.2-5).

$$f_1(t) \equiv B\,\Omega\cos\Omega t \tag{11.2.3.2-6}$$

The $f_2(t)$ term in (11.2.3.2-4) is developed from the integral of the square of (11.2.3.2-5):

$$\begin{aligned} f_2(t) &= \int \dot{\gamma}^2\,dt = \int \left(A + B\,\Omega\cos\Omega t\right)^2 dt \\ &= \int \left(A^2 + 2\,A\,B\,\Omega\cos\Omega t + B^2\,\Omega^2\cos^2\Omega t\right) dt \end{aligned} \tag{11.2.3.2-7}$$

or, after integration:

$$f_2(t) = \left(A^2 + \frac{1}{2}B^2\,\Omega^2\right) t + 2\,A\,B\sin\Omega t + \frac{1}{2}B^2\,\Omega\sin\Omega t\cos\Omega t \tag{11.2.3.2-8}$$

The simulated strapdown angular rate sensor triad output is the integral of $\underline{\omega}_{IB}^{B}$ over a sensor assembly sample period. Using (11.2.3.1-10) we find:

$$\Delta\underline{\alpha}_l = \int_{t_{l-1}}^{t_l} \underline{\omega}_{IB}^{B}\,dt = \int_{t_{l-1}}^{t_l} \dot{\gamma}\,\underline{u}_{\gamma}^{B}\,dt \tag{11.2.3.2-9}$$

or

$$\Delta\underline{\alpha}_l = \left(\gamma(t_l) - \gamma(t_{l-1})\right)\underline{u}_{\gamma}^{B} \tag{11.2.3.2-10}$$

where

$\Delta\underline{\alpha}_l$ = Strapdown angular rate sensor triad output vector over the l^{th} sensor assembly sample period.

The $\gamma(t)$ parameter in Equation (11.2.3.2-10) is given by Equation (11.2.3.1-11).

The $\Delta\underline{\alpha}_l$ and $\Delta\underline{\upsilon}_l$ simulated inertial sensor outputs given by Equations (11.2.3.2-10) and (11.2.3.2-3) (with (11.2.3.2-4), (11.2.3.2-6) and (11.2.3.2-8)) are used to represent the strapdown sensor inputs to the strapdown attitude integration, accelerometer size effect, acceleration transformation, velocity update and position determination algorithms under test (e.g., for the Chapter 7, 8 and 19 (Section 19.1) algorithms, $\Delta\underline{\alpha}_l$ and $\Delta\underline{\upsilon}_l$ in Equations (7.1.1.1.1-17), (7.2.2.2.2-14), (7.2.2.2.2-15), (7.3.3.2-18) - (7.3.3.2-20), (8.1.4.1.4-5) (with $\Delta\underline{\alpha}'_l$ set to $\Delta\underline{\alpha}_l$), and (19.1.11-1)).

11.2.3.3 REFERENCE ATTITUDE, VELOCITY AND POSITION

The Spin-Rock-Size attitude output is the orientation of the B Frame relative to a non-rotating attitude reference coordinate frame, or in terms of direction cosines:

$$C_B^L = C_{B_0}^L \, C_B^{B_0} \tag{11.2.3.3-1}$$

where

L = Non-rotating attitude reference coordinate frame.

C_B^L = Direction cosine matrix that transforms vectors from the B Frame to the L Frame.

$C_{B_0}^L$ = C_B^L at time $t = 0$.

$C_B^{B_0}$ = Direction cosine matrix that transforms vectors from the B Frame at time t to the B Frame attitude at time $t = 0$.

The $C_B^{B_0}$ matrix in (11.2.3.3-1) is a function of the angular position γ of the sensor assembly since $t = 0$ around the rotation axis $\underline{u}_\gamma^B$ or, considering $\gamma \, \underline{u}_\gamma^B$ as a rotation vector defining the orientation of the B Frame at time t relative to B at $t = 0$, from generalized Equation (3.2.2.1-4) we can write:

$$C_B^{B_0} = I + \sin\gamma \left(\underline{u}_\gamma^B \times\right) + \left(1 - \cos\gamma\right)\left(\underline{u}_\gamma^B \times\right)^2 \tag{11.2.3.3-2}$$

with γ given by Equation (11.2.3.1-11).

User specification of the $C_{B_0}^L$ matrix is easily accomplished by definition of three Euler angles describing the B Frame attitude relative to the L Frame at time $t = 0$ and then applying Equations (11.2.2.1-18) for $C_{B_0}^L$. The roll, pitch and heading Euler angles associated with C_B^L are evaluated as in Equations (11.2.1.3-2).

The reference velocity output from the Spin-Rock-Size simulator is defined as the time rate of change of the sensor assembly navigation center position as projected on the axes of a non-rotating coordinate frame. Analytically, the velocity can be defined as the rate of change of the $\underline{l}_0$ displacement vector defined in Section 11.2.3.1 in the non-rotating I Frame:

$$\underline{v}^I \equiv \dot{\underline{l}}_0^I \tag{11.2.3.3-3}$$

where

$\underline{v}^{I}$ = Navigation center velocity in the I Frame.

The (11.2.3.3-3) I Frame derivative is given by Equation (11.2.3.1-5) with $\underline{l}^{B} = \underline{l}_{0}^{B}$. Transforming (11.2.3.3-3) with Equation (11.2.3.1-5) and $\underline{l}^{B} = \underline{l}_{0}^{B}$ into the body B Frame gives the velocity vector in sensor assembly B Frame coordinates:

$$\underline{v}^{B} = \underline{\omega}_{IB}^{B} \times \underline{l}_{0}^{B} \tag{11.2.3.3-4}$$

where

$\underline{v}^{B}$ = Navigation center velocity in the B Frame.

In a non-rotating navigation reference coordinate frame, Equation (11.2.3.3-4) can be written as:

$$\underline{v}^{N} = C_{L}^{N} \, C_{B}^{L} \left(\underline{\omega}_{IB}^{B} \times \underline{l}_{0}^{B} \right) \tag{11.2.3.3-5}$$

where

$\underline{v}^{N}$ = Navigation center velocity in the N Frame.

N = Non-rotating navigation reference coordinate frame.

C_{L}^{N} = Constant direction cosine matrix that transforms vectors from the L Frame to the N Frame.

The C_{L}^{N} matrix in the Spin-Rock-Size simulator is defined as in Section 4.1.1 (based on the Section 2.2 definitions for Frames L and N) as the transpose of Equation (4.1.1-2):

$$C_{L}^{N} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{11.2.3.3-6}$$

Using (11.2.3.1-10) for $\underline{\omega}_{IB}^{B}$, Equation (11.2.3.3-5) becomes:

$$\underline{v}^{N} = \dot{\gamma} \, C_{L}^{N} \, C_{B}^{L} \left(\underline{u}_{\gamma}^{B} \times \underline{l}_{0}^{B} \right) \tag{11.2.3.3-7}$$

with $\dot{\gamma}$ given by (11.2.3.2-5).

The navigation center position in the N Frame is simply the $\underline{l}_0$ vector projected onto the N Frame:

$$\underline{R}^{N} = C_{L}^{N} \, C_{B}^{L} \, \underline{l}_{0}^{B} \tag{11.2.3.3-8}$$

where

$\underline{R}^N$ = Navigation center position vector in the N Frame.

11.2.3.4 STRAPDOWN ALGORITHM ERROR EVALUATION

The accuracy of the strapdown software attitude integration, accelerometer size effect compensation, acceleration transformation, velocity update, position determination algorithms is evaluated with the Spin-Rock-Size simulator by comparing the Spin-Rock-Size attitude matrix C_B^L and Euler angles, velocity $\underline{v}^N$ and position $\underline{R}^N$ with the equivalent strapdown software algorithm generated parameters at selected time points t.

The Spin-Rock-Size simulator C_B^L matrix and associated Euler angles at time t (from Equations (11.2.3.1-11), (11.2.3.3-1) - (11.2.3.3-2), (11.2.2.1-18) and (11.2.1.3-2)) are compared with the C_B^L matrix and Euler angles calculated using the strapdown software attitude integration routines to assess the strapdown algorithm attitude accuracy. Section (11.2.1.4) describes how the comparison is made to evaluate strapdown algorithm C_B^L error in terms of normality, orthogonality and misalignment error parameters.

The Spin-Rock-Size simulator velocity vector $\underline{v}^N$ at time t, provided by Equation (11.2.3.3-7) with (11.2.3.2-5), (11.2.3.3-6) and C_B^L (calculated per the previous paragraph), is compared against the equivalent parameter calculated with the strapdown software algorithms under test using the attitude integration, accelerometer size effect compensation, specific force acceleration transformation, and velocity update operations. For example, for the Chapter 7 algorithms, $\underline{v}^N$ would be calculated using the strapdown algorithm Equation (7.2-2) integration routine but with the gravity/Coriolis term set to zero. The result can then be compared with the Spin-Rock-Size generated $\underline{v}^N$ to evaluate the strapdown algorithm accuracy in generating integrated specific force acceleration in N Frame coordinates from the Spin-Rock-Size simulated inertial sensor inputs. Similarly, for the Chapter 19 (Section 19.1) unified algorithms, $\underline{v}^N$ would be calculated using the Equations (19.1.5-9) and (19.1.11-1) integration routines with the $\Delta\underline{v}_g^N$ term set to zero. The result would then be compared with the Spin-Rock-Size generated $\underline{v}^N$ to evaluate the algorithm accuracy in generating integrated specific force acceleration in N Frame coordinates from the Spin-Rock-Size simulated inertial sensor inputs.

The Spin-Rock-Size simulator position vector $\underline{R}^N$ at time t provided by Equation (11.2.3.3-8) is compared against the equivalent parameter calculated with the strapdown software algorithms under test from the attitude integration, accelerometer size effect compensation, specific force acceleration transformation, velocity update and position change increment operations. For example, for the Chapter 7 algorithms, $\underline{v}^N$ would be calculated using the strapdown algorithm Equation (7.2-2) integration routine with the gravity/Coriolis term set to zero and (for a high precision position updating routine) the $\Delta\underline{R}_m^N$ N Frame position change increment would be calculated using Equation (7.3.3-8), but with the gravity, Coriolis and $\underline{\zeta}_{n-1,m}$, $\underline{\zeta}_{n-1,m-1}$ L Frame rotation terms set to zero, and $C_{L(n-1)}^{L(m-1)}$ set to identity. The $\Delta\underline{R}_m^N$ increment would then be summed at the m cycle transformation rate to obtain the equivalent to $\underline{R}^N$ for Spin-Rock-Size comparison. If a standard trapezoidal position integration algorithm is being used in the software under test (rather than the high resolution version), the equivalent $\Delta\underline{R}_m^N$ increment would be calculated using Equation (7.3.2-1), with the result then summed to obtain $\underline{R}^N$. The Chapter 19 Equations (19.1.3-2) unified approach position algorithms can also be validated with the Spin-Rock-Size simulator using $\underline{R}^N$ from (19.1.3-2) with (19.1.11-1) compared to $\underline{R}^N$ from the simulator. The $\underline{v}^N$ input to $\underline{R}^N$ in (19.1.3-2) would be calculated with $\Delta\underline{v}_g^N$ set to zero.

11.2.4 GEN NAV SIMULATOR

The Gen Nav simulator provides a complete set of analytical closed-form navigation data at user selected time points including a simulated strapdown angular rate sensor and accelerometer output history whose processing in exact strapdown integration algorithms would generate the identically same navigation data. The Gen Nav simulator is utilized to verify strapdown inertial navigation software algorithms under test by operating the algorithms using the Gen Nav simulator strapdown inertial sensor outputs and then comparing the navigation data generated by the algorithms with the equivalent data provided by the Gen Nav simulator.

The Gen Nav simulator is structured around a user specified latitude/longitude/altitude position profile that is analytically time differentiable so that the associated rates of change are also analytic functions. The specified latitude/longitude/altitude functions (and their rates of change) are used to calculate the position range vector from earth’s center, velocity and gravity data in an inertial (non-rotating) coordinate frame. The computed velocity and gravity data are then combined to calculate incremental changes in integrated specific force acceleration in the inertial frame. An angular rate profile is also user specified for the strapdown sensor coordinate axes (“body” B Frame), which is used to form the Gen Nav simulated strapdown angular rate sensor output signals and to calculate B Frame attitude. The B Frame attitude is used to transform the inertial integrated specific force acceleration increments to body frame axes to generate the Gen Nav simulated strapdown accelerometer outputs. For the above operations, analytically exact routines are used for the earth shape model and integration routines so that Gen Nav outputs can be used as an accurate reference for the validation of high accuracy strapdown inertial navigation software algorithms. The gravity model is selected to match that used in the software algorithms under test.

The sections to follow present the analytical basis for the Gen Nav simulator computation routines. Section 11.2.4.1 (and its subsections) defines the user specified latitude/longitude/altitude analytic position parameters (and their analytic derivatives) used to describe the reference trajectory profile, and converts these into position-range/velocity vector data in inertial coordinates. Section 11.2.4.2 provides equations for converting the Section 11.2.4.1 position/velocity data into earth referenced latitude/longitude/altitude position location and north/east/vertical velocity outputs. Section 11.2.4.3 and its subsections derive expressions for the simulated strapdown inertial sensor outputs. Section 11.2.4.4 describes the equations used to provide roll, pitch, true heading outputs.

In the subsections to follow we will be using the B, I, and E Frames as defined in Section 2.2 and where for more specificity:

I = Inertial non-rotating coordinate frame having the Y-axis along earth's polar axis, X, Z axes parallel to the earth's equatorial plane, and the Z-axis in the plane of the earth Greenwich longitude reference meridian at time t = 0.

E = Earth fixed coordinate frame having Y along the earth polar axis, Z, X in the earth equatorial plane, with Z in the Greenwich England meridian plane. Note that from the previous definition for the I Frame, that the I and E Frames are coincident at simulation time t = 0.

11.2.4.1 POSITION AND VELOCITY PARAMETERS

The latitude/longitude/altitude input profile is specified in terms of analytic functions with analytic time derivatives (i.e., rates of change of latitude/longitude/altitude) relative to a non-rotating (inertial) geoid surface that is of identical shape to the ellipsoidal earth surface reference geoid (See Section 5.1 for earth shape definition). Thus, the non-rotating inertial geoid is tangent at all points to the rotating earth surface. From this definition, and the definitions for the E and I Frames in Section 11.2.4, it should be clear that latitude and altitude relative to the inertial geoid are identical to latitude/altitude relative to the rotating earth referenced geoid (i.e., desired Gen Nav outputs). Earth referenced longitude (for output), however, equals the inertially defined longitude plus the rotation angle of the earth about its polar axis since simulation start. The following subsections describe the analytical functions used in the Gen Nav simulator for inertial latitude/longitude/altitude (and their derivatives), and their conversion equations into the inertial frame position-range and velocity vectors.

11.2.4.1.1 Specified Latitude/Inertial-Longitude/Altitude And Their Derivatives

In principle, the latitude/inertial-longitude/altitude input parameters can be any analytically differentiable time functions. For the Gen Nav simulator, they are selected to approximate a general aircraft trajectory profile.

The horizontal position functions (latitude, inertial longitude) are each represented as an offset negative cosine wave for one quarter of a cosine wave cycle, followed by a constant linear-with-time ramp that matches the negative cosine function amplitude at the cosine function first quarter cycle point. The ramp slope is selected to match the derivative of the negative cosine function at the ramp start time (at the cosine function quarter cycle time). In the case of the longitude function, the cosine/ramp function is augmented by an additional linear time-ramp initiated at run time zero. By setting the slope of the additional ramp to earth's rotation rate, the specified inertial longitude negative-cosine/ramp function (relative to the inertial geoid) simulates motion relative to the rotating earth. Having specified analytical functions for the latitude and inertial longitude, their associated rates of change are easily calculated by analytical differentiation. Thus, we write:

$$
\begin{aligned}
&\text{If } (\omega_l\, t < \pi / 2), \text{ Then:} \\
&\qquad l = l_0 + l_{Amp}\,(1 - \cos\, \omega_l\, t) \\
&\qquad \dot{l} = l_{Amp}\, \omega_l \sin \omega_l\, t \\
&\text{If } (\omega_l\, t \geq \pi / 2), \text{ Then:} \\
&\qquad l = l_0 + l_{Amp}\,(1 + \omega_l\, t - \pi / 2) \\
&\qquad \dot{l} = l_{Amp}\, \omega_l \\
&\text{If } (\omega_L\, t < \pi / 2), \text{ Then:} \\
&\qquad L_I = L_0 + L_1\, t + L_{Amp}\,(1 - \cos\, \omega_L t) \\
&\qquad \dot{L}_I = L_1 + L_{Amp}\, \omega_L \sin \omega_L t \\
&\text{If } (\omega_L\, t \geq \pi / 2), \text{ Then:} \\
&\qquad L_I = L_0 + L_1\, t + L_{Amp}\,(1 + \omega_L\, t - \pi / 2) \\
&\qquad \dot{L}_I = L_1 + L_{Amp}\, \omega_L
\end{aligned}
\qquad (11.2.4.1.1\text{-}1)
$$

where

t = Time from simulator run start (sec).

l = Latitude relative to the inertial (and earth reference) geoid (rad).

l_0 = Initial latitude (rad).

l_{Amp} = Latitude cosine function amplitude (rad).

ω_l = Frequency of latitude cosine function (rad/sec).

$\dot{l}$ = Rate of change of latitude (rad/sec).

L_I = Inertial longitude relative to the inertial geoid (rad).

L_0 = Initial longitude (rad). Because the E and I Frames have been defined to be coincident at time $t = 0$, L_0 is the initial value for both the inertial longitude L_I and earth referenced longitude relative to Greenwich.

L_1 = Inertial longitude offset time ramp slope constant (rad/sec).

L_{Amp} = Inertial longitude cosine function amplitude (rad).

ω_L = Frequency of inertial longitude cosine function (rad/sec).

$\dot{L}_I$ = Rate of change of inertial longitude (rad/sec).

The altitude function is defined as a half wave of an offset negative cosine function followed by a second offset negative cosine function (at a different frequency) that matches the amplitude and slope of the first at its first half wave point. The altitude rate is the analytical derivative of the altitude function. Thus:

$$
\begin{aligned}
&\text{If } (\omega_{h1}\, t < \pi), \text{ Then:} \\
&\qquad h = h_0 + h_{Amp1}\,(1 - \cos\,\omega_{h1}\, t) \\
&\qquad \dot{h} = h_{Amp1}\,\omega_{h1} \sin\omega_{h1}\, t \\
&\text{If } (\omega_{h1}\, t \geq \pi), \text{ Then:} \\
&\qquad h = h_0 + 2\,h_{Amp1} + h_{Amp2}\left(1 - \cos\,\omega_{h2}(t - \pi / \omega_{h1})\right) \\
&\qquad \dot{h} = h_{Amp2}\,\omega_{h2} \sin\omega_{h2}(t - \pi / \omega_{h1})
\end{aligned}
\qquad (11.2.4.1.1\text{-}2)
$$

where

h = Altitude above the inertial (and earth reference) geoid surface.

h_0 = Initial altitude (ft).

h_{Amp1} = First altitude cosine function amplitude (ft).

ω_{h1} = Frequency of first altitude cosine function (rad/sec).

h_{Amp2} = Second altitude cosine function amplitude (ft).

ω_{h2} = Frequency of second altitude cosine function (rad/sec).

$\dot{h}$ = Rate of change of altitude (ft/sec).

Values for the coefficient parameters in (11.2.4.1.1-1) and (11.2.4.1.1-2) are set to simulate typical aircraft profile characteristics. From the form of the latitude and altitude equations, we see that the initial north/vertical velocity (i.e., latitude/altitude rate) are zero at time $t = 0$. The L_1 coefficient in (11.2.4.1.1-1) is set equal to earth's inertial angular rotation rate. The effect is to simulate zero east velocity relative to the rotating earth at time zero (which corresponds to a longitude rate over the non-rotating inertial geoid equal to earth's rotation rate). Thus:

$$L_1 = \omega_e \tag{11.2.4.1.1-3}$$

where

ω_e = Earth's inertial rotation rate (rad/sec).

It is important to recognize (as discussed in Section 11.2.4.1) that by setting L_1 to earth's rate in (11.2.4.1.1-3), the initial longitude (L_0) and cosine function in Equations (11.2.4.1.1-1) then represents the longitude profile relative to the <u>rotating</u> earth (as does the latitude function in (11.2.4.1.1-1) and the altitude function in (11.2.4.1.1-2)). Visualization of the resulting trajectory profile relative to the rotating earth is thereby simplified.

The l_{Amp} and L_{Amp} coefficients are set so that the latitude/inertial-longitude rates ($\dot{l}$ and $\dot{L}_I$) at the first quarter wave cosine function time points correspond to typical application linear horizontal cruise velocity component magnitudes (e.g., 1000 fps for an aircraft). Neglecting earth's ellipticity, we see from Equations (4.4.3-2), (4.4.3-5) and (4.4.3-6) that latitude/longitude rates relative to the earth are in terms of north/east velocity components:

$$\dot{l} = v_Y \,/\, (R_0 + h) \qquad \dot{L} = v_X \,/\, \big((R_0 + h)\cos\, l\big) \tag{11.2.4.1.1-4}$$

where

L = Longitude relative to Greenwich in the earth fixed E Frame.

v_X, v_Y = East/north velocity relative to the earth.

R_0 = Earth's equatorial radius.

We also know from previous paragraphs that:

$$\dot{L} = \dot{L}_I - \omega_e \tag{11.2.4.1.1-5}$$

Then, using (11.2.4.1.1-4) - (11.2.4.1.1-5) at the first quarter cycle cosine function time points, we see from (11.2.4.1.1-1) with (11.2.4.1.1-3) that:

$$\begin{aligned} l_{Amp} &= V_{North} \,/\, \Big((R_0 + h_{@t=\pi/(2\omega_l)})\ \omega_l\Big) \\ L_{Amp} &= V_{East} \,/\, \Big((R_0 + h_{@t=\pi/(2\omega_L)})\ \omega_L \cos\, l_{@t=\pi/(2\omega_L)}\Big) \end{aligned} \tag{11.2.4.1.1-6}$$

where

V_{North} , V_{East} = North and East cruise velocities (in fps) at the first quarter cycle cosine function time points (i.e., at $t = \pi / (2\, \omega_l)$ for V_{North} and $t = \pi / (2\, \omega_L)$ for V_{East}).

$@t=(\,)$ = Subscript designation for parameter evaluated from Equations (11.2.4.1.1-1) - (11.2.4.1.1-2) at the designated time points.

The frequencies for the l and L_I cosine functions are set so that the first quarter wave end points correspond to a time that is typical for the application (i.e., 900 sec for an airplane to reach cruising altitude, corresponding to a full sine wave period of 3600 sec):

$$\omega_l = 2\,\pi / T_0 \qquad \omega_L = 2\,\pi / T_0 \tag{11.2.4.1.1-7}$$

where

T_0 = Time period for the latitude/inertial-longitude cosine wave functions.

The h_{Amp1} coefficient for the altitude function (Equations (11.2.4.1.1-2)) is set so that at the first cosine half wave end point, h equals a typical application cruise altitude (e.g., 40,000 feet for an aircraft). From (11.2.4.1.1-2), the associated initialization equation is:

$$h_{Amp1} = \frac{1}{2}\,(H_{Cruise} - h_0) \tag{11.2.4.1.1-8}$$

where

H_{Cruise} = Selected nominal cruise altitude (ft).

The first altitude cosine function frequency is set so that the half wave end point corresponds to the first quarter wave point for the latitude/inertial-longitude cosine functions. With (11.2.4.1.1-7) this is given by:

$$\omega_{h1} = 4\,\pi / T_0 \tag{11.2.4.1.1-9}$$

The second altitude cosine function amplitude and frequency are set to provide a selected oscillatory altitude during cruise that is above (or below) the nominal cruise altitude. From (11.2.4.1.1-2):

$$h_{Amp2} = \frac{1}{2}\,\Delta h_{Osc} \qquad \omega_{h2} = 2\,\pi / T_{hOsc} \tag{11.2.4.1.1-10}$$

where

Δh_{Osc} = Peak oscillatory altitude variation during cruise above (below) the nominal cruise altitude.

T_{hOsc} = Period for the oscillatory altitude variation during cruise.

11.2.4.1.2 Inertial Position Range And Velocity Vectors

We can use Equations (4.4.2.1-2) as representing a general set of relationships between latitude, longitude, wander angle Euler angles and the elements of a general direction cosine matrix $C_{A_2}^{A_1}$ relating generalized coordinate frames A_2 and A_1 (with elements D_{ij} as given in (4.4.2.1-2)). Recall that Equations (4.4.2.1-2) are based on the definition of the direction cosine matrix having the Z axis of the associated locally level navigation frame (the N Frame corresponding to A_2 of $C_{A_2}^{A_1}$) along the upward vertical. Generalized Equations (3.2.1-6) show that a column in a general direction cosine matrix $C_{A_2}^{A_1}$ (say column i) represents a unit vector along A_2 Frame axis i projected on A_1 Frame coordinate axes. Let us now consider Frame A_1 as being the inertial I Frame and Frame A_2 as having its Z axis along the upward vertical. Then the unit vector along the upward vertical has I Frame components given by the third column of the (4.4.2.1-2) direction cosine matrix. Thus:

$$\underline{u}_{Up}^{I} = \begin{bmatrix} u_{Up_{XI}} \\ u_{Up_{YI}} \\ u_{Up_{ZI}} \end{bmatrix} = \begin{bmatrix} \sin L_I \cos l \\ \sin l \\ \cos L_I \cos l \end{bmatrix} \tag{11.2.4.1.2-1}$$

where

$\underline{u}_{Up}^{I}$ = Unit vector along the upward local geodetic vertical as projected on I Frame axes.

$u_{Up_{XI}}, u_{Up_{YI}}, u_{Up_{ZI}}$ = I Frame X, Y, Z components of $\underline{u}_{Up}^{I}$.

The equations developed in Sections 5.1 and 5.2.2 of Chapter 5 describe navigation parameters in the earth fixed E Frame relative to an ellipsoidal earth referenced geoid surface. Since the inertial geoid has been defined to be tangent at all points to the earth referenced geoid, the equations developed in the previous sections can also be applied for I Frame defined parameters relative to the inertial geoid. Thus, from equations (5.1-10) and (5.2.2-1), we can quickly write for the position range vector:

$$R_S' = R_0 / \sqrt{1 + u_{Up_{YI}}^2 \left[(1 - e)^2 - 1 \right]} \tag{11.2.4.1.2-2}$$

$$\underline{R}^{I} = \begin{bmatrix} u_{Up_{XI}} \left(R_S' + h \right) \\ u_{Up_{YI}} \left[(1 - e)^2 R_S' + h \right] \\ u_{Up_{ZI}} \left(R_S' + h \right) \end{bmatrix} \tag{11.2.4.1.2-3}$$

where

R_S' = Modified distance parameter from earth's center to the geoid surface.

$\underline{R}^I$ = Position range vector from earth's center to the current position location in I Frame coordinates.

e = Ellipsoidal earth shape geoid model ellipticity.

R_0 = Earth's equatorial radius.

We also note that the linear distance from earth center to the current position location is the magnitude of $\underline{R}^I$ or:

$$R = \sqrt{\underline{R}^I \cdot \underline{R}^I} \tag{11.2.4.1.2-4}$$

where

R = Magnitude of $\underline{R}^I$.

The velocity relative to the non-rotating I Frame is the derivative of (11.2.4.1.2-3), hence:

$$\dot{\underline{R}}^I = \begin{bmatrix} \dot{u}_{Up_{XI}}\left(R_S' + h\right) + u_{Up_{XI}}\left(\dot{R}_S' + \dot{h}\right) \\ \dot{u}_{Up_{YI}}\left[(1 - e)^2 R_S' + h\right] + u_{Up_{YI}}\left[(1 - e)^2 \dot{R}_S' + \dot{h}\right] \\ \dot{u}_{Up_{ZI}}\left(R_S' + h\right) + u_{Up_{ZI}}\left(\dot{R}_S' + \dot{h}\right) \end{bmatrix} \tag{11.2.4.1.2-5}$$

The derivative terms in (11.2.4.1.2-5) are obtained by analytic differentiation of Equations (11.2.4.1.2-1) and (11.2.4.1.2-2):

$$\begin{bmatrix} \dot{u}_{Up_{XI}} \\ \dot{u}_{Up_{YI}} \\ \dot{u}_{Up_{ZI}} \end{bmatrix} = \begin{bmatrix} \dot{L}_I \cos L_I \cos l - \dot{l} \sin L_I \sin l \\ \dot{l} \cos l \\ - \dot{L}_I \sin L_I \cos l - \dot{l} \cos L_I \sin l \end{bmatrix} \tag{11.2.4.1.2-6}$$

$$\dot{R}_S' = - R_0\, u_{Up_{YI}} \left[(1 - e)^2 - 1\right] \dot{u}_{Up_{YI}} \left(1 + u_{Up_{YI}}^2 \left[(1 - e)^2 - 1\right]\right)^{-\frac{3}{2}} \tag{11.2.4.1.2-7}$$

in which $\dot{u}_{Up_{YI}}$ for (11.2.4.1.2-7) is provided by (11.2.4.1.2-6), and the $l, \dot{l}, L_I, \dot{L}_I, h, \dot{h}$ parameters in (11.2.4.1.2-5) and (11.2.4.1.2-6) are calculated by Equations (11.2.4.1.1-1) - (11.2.4.1.1-2). Equation (11.2.4.1.2-7) can be simplified by applying (11.2.4.1.2-2):

$$\dot{R}_S' = - u_{Up_{YI}} \left[(1 - e)^2 - 1\right] \frac{R_S'^{\,3}}{R_0^2}\, \dot{u}_{Up_{YI}} \tag{11.2.4.1.2-8}$$

In summary, the $\underline{R}^I$, R and $\dot{\underline{R}}^I$ vectors are calculated in the Gen Nav simulator through Equations (11.2.4.1.2-3) - (11.2.4.1.2-5) using (11.2.4.1.2-1), (11.2.4.1.2-2), (11.2.4.1.2-6) and (11.2.4.1.2-7), with l, $\dot{l}$, L_I, $\dot{L}_I$, h, $\dot{h}$ from Equations (11.2.4.1.1-1) - (11.2.4.1.1-2).

11.2.4.2 POSITION AND VELOCITY OUTPUT PARAMETERS

Section 11.2.4.1 showed how the position range and velocity vectors are calculated in the non-rotating I Frame from user defined latitude/inertial-longitude/altitude analytical models. For the Gen Nav simulator output, the equivalent data is required relative to the rotating earth. Position outputs are required in terms of latitude, longitude, altitude relative to rotating earth coordinates, and velocity outputs are required as the position rate of change relative to the rotating earth along local north, east, vertical axes.

As discussed in Section 11.2.4.1, earth referenced latitude and altitude are identical to the latitude/altitude (l, h) parameters defined analytically by the user in Equations (11.2.4.1.1-1) - (11.2.4.1.1-2), and longitude relative to the earth is computed from I Frame inertial longitude (L_I) (defined by the user in Equations (11.2.4.1.1-1)) by subtracting the angular rotation of the earth since Gen Nav simulator time $t = 0$.

Longitude relative to the earth E Frame is obtained from I Frame relative inertial longitude (L_I) by noting that the I and E Frames have identical Y axes, and that their X, Z axes are defined to be coincident at simulation start time $t = 0$. Thus, using Figure 4.4.2.1-1 for longitude definition, longitude in the E and I Frames will differ only by the angular displacement of the earth around its polar axis since time $t = 0$, and we can write:

$$L = L_I - \omega_e t \qquad (11.2.4.2\text{-}1)$$

where

L = Earth referenced longitude for Gen Nav simulator output.

From their fundamental definitions as Euler angles, latitude and longitude (l, L) can have multiple values for a given C_N^E matrix earth location (e.g., $l = \pi$, $L = \pi$ corresponds to the same C_N^E as $l = 0$, $L = \pi$). Because of the method for calculating l and L from (11.2.4.1.1-1), it is possible that the values may not fit the normal angle boundary constraints for these parameters (i.e., $|l| \leq \pi / 2$ and $|L| \leq \pi$). A straight-forward method for converting to the standard format is to first calculate the C_N^E direction cosine matrix from l, L using Equations (4.4.2.1-2) with zero wander angle (α), and then apply l, L extraction Equations (4.4.2.1-3) to the result. Since

Equations (4.4.2.1-3) are based on the normal l, L angle boundary definitions, the solution so obtained will provide l, L that meets the angle boundary constraints for the same C_N^E matrix.

North, east, and vertical components of velocity relative to the earth are calculated from $\dot{\underline{R}}^I$ using Equation (4.3-9), and projecting the resultant earth referenced velocity vector onto locally north, east, and vertical axes. Using (4.3-9) we write in the I Frame:

$$\underline{v}^I = \dot{\underline{R}}^I - \underline{\omega}_e^I \times \underline{R}^I \qquad \underline{\omega}_e^I = \omega_e \, \underline{u}_{YI}^I$$
$$v_{East} = \underline{v}^I \cdot \underline{u}_{East}^I \qquad v_{North} = \underline{v}^I \cdot \underline{u}_{North}^I \qquad v_{Up} = \underline{v}^I \cdot \underline{u}_{Up}^I \tag{11.2.4.2-2}$$

where

$\underline{v}^I$ = Velocity relative to the earth projected on I Frame axes.

$\underline{\omega}_e^I$ = Earth rotation rate vector relative to inertial space projected on I Frame axes.

$\underline{u}_{YI}^I$ = Unit vector along the I Frame Y axis (i.e., the earth polar axis).

$\underline{u}_{East}^I, \underline{u}_{North}^I, \underline{u}_{Up}^I$ = Unit vectors along local East, North, Up directions projected on I Frame axes.

$v_{East}, v_{North}, v_{Up}$ = Local East, North, Up components of $\underline{v}^I$. Outputs from the Gen Nav simulator.

The I Frame X, Y, Z components of $\underline{u}_{Up}^I$ are evaluated with (11.2.4.1.2-1) using (11.2.4.1.1-1) for l and L_I.

The $\underline{u}_{East}^I$, $\underline{u}_{North}^I$ vectors in (11.2.4.2-2) are calculated from the I Frame version of Equations (5.4.1-6).

$$\underline{u}_{East}^I = \frac{1}{\sqrt{1 - u_{Up\,YI}^2}} \begin{bmatrix} u_{Up\,ZI} \\ 0 \\ - u_{Up\,XI} \end{bmatrix} \qquad \underline{u}_{North}^I = \frac{1}{\sqrt{1 - u_{Up\,YI}^2}} \begin{bmatrix} - u_{Up\,XI}\, u_{Up\,YI} \\ 1 - u_{Up\,YI}^2 \\ - u_{Up\,ZI}\, u_{Up\,YI} \end{bmatrix} \tag{11.2.4.2-3}$$

From the definition of the I Frame axes in Section 11.2.4 we can write for $\underline{u}_{YI}^I$ in (11.2.4.2-2):

$$\underline{u}_{YI}^I = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{11.2.4.2-4}$$

11.2.4.3 SIMULATED STRAPDOWN INERTIAL SENSOR OUTPUTS

The simulated strapdown inertial sensor outputs from the Gen Nav simulator are based on constant body B Frame integrated angular rate and specific force acceleration increments over each attitude-update/acceleration-transformation cycle. Selection of the B Frame integrated angular rate increment profile (i.e., simulated strapdown angular rate sensor outputs) is user specified. The B Frame integrated specific force acceleration increment profile (i.e., simulated strapdown accelerometer outputs) is calculated based on the B Frame attitude profile associated with the selected angular rate profile, the derivative of the computed inertial position range rate ($\dot{\underline{R}}^I$), and the Gen Nav simulator gravity model. The details are provided in the following subsections.

11.2.4.3.1 Simulated Strapdown Angular Rate Sensor Outputs

The Gen Nav simulator angular rate profile is based on a user specified inertial attitude change profile for the sensor assembly over the acceleration-transformation/attitude-update cycle (m) of the software under test. The inertial attitude change over the m cycle is represented by a B Frame body rotation angle vector ($\underline{\phi}_m$). The Gen Nav simulator assumes a constant inertial angular rate over each m cycle. Thus, according to Equation (7.1.1.1-14), the user specified $\underline{\phi}_m$ profile represents integrated inertial angular rate increments, hence, simulated strapdown angular rate sensor output increments. Any easily definable function will suffice for $\underline{\phi}_m$. For example, the $\underline{\phi}_m$ vector can be set to a constant for 100 acceleration-transformation/attitude-update cycles beginning at t = 200 seconds, and zero at all other times. This is easily achieved by setting the $\underline{\phi}_m$ vector magnitude (ϕ_m) and direction vector ($\underline{u}_\phi$) to user selected constants (e.g., setting the magnitude ϕ_m to 0.02 radians, the X, Y components of the direction vector $\underline{u}_\phi$ to +0.3, +0.55, and the Z component to $\pm\sqrt{1 - 0.3^2 - 0.55^2}$ to assure unity $\underline{u}_\phi$ magnitude).

The B Frame attitude corresponding to the selected $\underline{\phi}_m$ profile is calculated in the Gen Nav simulator utilizing the classical form given by generalized Equation (3.2.2.1-4) with the (3.2.1-5) chain rule:

$$
\begin{aligned}
C^I_{B_m} &= C^I_{B_{m-1}} \; C^{B_{m-1}}_{B_m} \\
C^{B_{m-1}}_{B_m} &= I + \sin \phi_m \left(\underline{u}_\phi \times\right) + (1 - \cos \phi_m) \left(\underline{u}_\phi \times\right)^2
\end{aligned}
\qquad (11.2.4.3.1\text{-}1)
$$

where

C^I_B = Direction cosine matrix that transforms vectors from B to I Frame coordinates.

11.2.4.3.1.1 Attitude Matrix Initialization

The C_B^I matrix in Equations (11.2.4.3.1-1) is initialized based on input starting roll, pitch, heading relative to starting local East, North, Up geographic coordinates (Geo Frame) and the starting orientation of the Geo Frame relative to the I Frame:

$$C_{B_0}^{I} = C_{Geo_0}^{I} \, C_{B_0}^{Geo_0} \tag{11.2.4.3.1.1-1}$$

where

Geo = East, North, Up coordinates.

0 = Starting (initial) value.

The $C_{B_0}^{Geo_0}$ matrix in (11.2.4.3.1.1-1) is calculated in terms of initial roll, pitch, true heading using Equations (11.2.2.1-18) for $C_{B_0}^{Geo_0}$ premultiplied by C_L^N. The $C_{Geo_0}^{I}$ matrix in (11.2.4.3.1.1-1) is obtained from the Section 11.2.4 definitions for the I and E Frames being coincident at simulation time $t = 0$, Equations (4.4.2.1-2) for the C_N^E matrix with the wander angle (α) set to zero (to make the N Frame equivalent to the Geo Frame), and the initial latitude/longitude (l_0, L_0) set as used in (11.2.4.1.1-1):

$$C_{Geo_0}^{I} = \begin{bmatrix} \cos L_0 & -\sin L_0 \sin l_0 & \sin L_0 \cos l_0 \\ 0 & \cos l_0 & \sin l_0 \\ -\sin L_0 & -\cos L_0 \sin l_0 & \cos L_0 \cos l_0 \end{bmatrix} \tag{11.2.4.3.1.1-2}$$

11.2.4.3.2 <u>Simulated Strapdown Accelerometer Outputs</u>

The simulated strapdown accelerometer outputs from the Gen Nav simulator are calculated from I Frame integrated specific force acceleration increments converted to sensor assembly B Frame coordinates.

11.2.4.3.2.1 Integrated I Frame Specific Force Acceleration Increments

Specific force acceleration is calculated in the Gen Nav simulator as integrated specific force acceleration increments over the acceleration-transformation m cycle for the software under test. The integrated specific force acceleration increment is computed in the I Frame as the change in I Frame position rate ($\dot{\underline{R}}^I$) over an m cycle, minus the integral of local gravity over the m cycle. A trapezoidal algorithm is used for the gravity integrated increment. Symbolically:

$$\Delta \underline{v}_{SF_m}^{I} = \dot{\underline{R}}_m^I - \dot{\underline{R}}_{m-1}^I - \frac{1}{2}\left(\underline{g}_m^I + \underline{g}_{m-1}^I\right) T_m \qquad (11.2.4.3.2.1\text{-}1)$$

where

m = Acceleration transformation cycle time index.

$\Delta \underline{v}_{SF_m}^{I}$ = Integrated specific force acceleration in the I Frame over the m-1 to m transformation cycle time.

T_m = Time interval from m-1 to m.

$\underline{g}_m^I$ = Gravity vector in I Frame coordinates at time t_m.

The $\dot{\underline{R}}^I$ terms in (11.2.4.3.2.1-1) are obtained from Equation (11.2.4.1.2-5). The $\underline{g}_m^I$ term is selected in the Gen Nav simulator to match the gravity model used in the software under test. If the Section 5.4 gravity model is used (as developed for positive altitude h in Reference 3 - Section 4.4 and Reference 4, and in Section 5.4 for negative altitude) the components would be as summarized in Equations (5.4-1) and (5.4-2) repeated below:

For $h \geq 0$:

$$g_r = -\frac{\mu}{R^2}\left[1 - \frac{3}{2} J_2 \left(\frac{R_0}{R}\right)^2 \left(3 \cos^2 \phi - 1\right) - 2 J_3 \left(\frac{R_0}{R}\right)^3 \cos \phi \left(5 \cos^2 \phi - 3\right) - \cdots\right]$$

$$\left(\frac{g_\phi}{\sin \phi}\right) = 3 \frac{\mu}{R^2}\left(\frac{R_0}{R}\right)^2 \left[J_2 \cos \phi + \frac{1}{2} J_3 \frac{R_0}{R}\left(5 \cos^2 \phi - 1\right) + \cdots\right]$$

$$g_\theta \approx 0 \qquad (11.2.4.3.2.1\text{-}2)$$

For $h < 0$:

$$g_r = \frac{R}{R_S} g_{r_S} \qquad \left(\frac{g_\phi}{\sin \phi}\right) = \frac{R}{R_S}\left(\frac{g_\phi}{\sin \phi}\right)_S \qquad g_\theta \approx 0$$

where

μ = Product of the mass of the earth with the universal gravitational constant.

J_2, J_3, etc. = Empirical constants that are a function of the mass distribution of the earth.

ϕ = Angle from the earth polar axis to $\underline{R}$ (see Figure 5.2-1).

g_r = Component of gravity along the $\underline{R}$ direction.

g_ϕ = Component of gravity perpendicular to $\underline{R}$ in the local meridian plane (positive in the positive ϕ direction).

g_θ = Component of gravity perpendicular to $\underline{R}$ and perpendicular to the local meridian plane.

R = Linear distance from earth's center to the current position location.

$g_{r_S}, \left(\dfrac{g_\phi}{\sin\phi}\right)_S$ = Values for g_r and $\left(\dfrac{g_\phi}{\sin\phi}\right)$ calculated from the top set in Equations (11.2.4.3.2.1-2) (i.e., for $h \geq 0$), and with R set to R_S defined below.

R_S = Distance from the center of the earth to the point on the earth geoid surface that is directly below (above) the $\underline{R}$ position vector location. The R_S earth surface point is defined such that a line from it to the $\underline{R}$ location point is perpendicular to a plane tangent to the earth's surface at the R_S surface point.

The R parameter in (11.2.4.3.2.1-2) is provided from Equation (11.2.4.1.2-4), and R_S is provided by Equation (5.2.1-4) (repeated below with $u_{Up_{YE}}$ equated to $u_{Up_{YI}}$ as in (11.2.4.1.2-2)) using (11.2.4.1.2-2) for R_S':

$$R_S = R_S' \sqrt{1 + u_{Up_{YI}}^2 \left[(1 - e)^4 - 1\right]} \tag{11.2.4.3.2.1-3}$$

The equivalent gravity components in earth's equatorial plane and along earth's polar axis (See Figure 5.2-1) are obtained by the transformation of g_r, g_ϕ components through the ϕ angle:

$$g_{EPA} = g_r \cos\phi - g_\phi \sin\phi \qquad g_{Eq} = g_r \sin\phi + g_\phi \cos\phi \tag{11.2.4.3.2.1-4}$$

where

g_{EPA}, g_{Eq} = Components of gravity along earth's polar axis and in the equatorial plane. Note that from the symmetry assumed for the Section 5.4 gravity model around earth's polar axis, g_{Eq} is in the local meridian plane, and there is no gravity component perpendicular to the local meridian plane.

Equations (11.2.4.3.2.1-4) can be rearranged to avoid singularities at high/low latitudes in subsequent processing equations:

$$\begin{aligned} g_{EPA} &= g_r \cos\phi - \left(\frac{g_\phi}{\sin\phi}\right)\left(\frac{\sin\phi}{\sqrt{1 - u_{Up_{YI}}^2}}\right)^2 \left(1 - u_{Up_{YI}}^2\right) \\ \left(\frac{g_{Eq}}{\sqrt{1 - u_{Up_{YI}}^2}}\right) &= g_r \left(\frac{\sin\phi}{\sqrt{1 - u_{Up_{YI}}^2}}\right) + \left(\frac{g_\phi}{\sin\phi}\right)\left(\frac{\sin\phi}{\sqrt{1 - u_{Up_{YI}}^2}}\right)\cos\phi \end{aligned} \tag{11.2.4.3.2.1-5}$$

The g_r and $\left(g_\phi / \sin\phi\right)$ terms in (11.2.4.3.2.1-5) are provided by (11.2.4.3.2.1-2) and $u_{Up_{YI}}$ is obtained from Equations (11.2.4.1.2-1). The $\left(\sin\phi / \sqrt{1 - u_{Up_{YI}}^2}\right)$ parameter in (11.2.4.3.2.1-5) is evaluated using Equation (5.2.2-5) (repeated below):

$$\left(\frac{\sin\phi}{\sqrt{1 - u_{Up_{YI}}^2}}\right) = \left(R_S' + h\right) / R \qquad (11.2.4.3.2.1\text{-}6)$$

in which it is recognized that the I and E Frame Y axes are coincident, hence, $u_{Up_{YE}} = u_{Up_{YI}}$. The cos ϕ term in (11.2.4.3.2.1-5) is evaluated from the definition of ϕ in Figure 5.2-1:

$$\cos\phi = R_{YI} / R \qquad (11.2.4.3.2.1\text{-}7)$$

where

R_{YI} = I Frame Y axis component of $\underline{R}^I$ provided by Equation (11.2.4.1.2-3).

The geodetic vertical and north components of gravity are calculated based on the g_{EPA}, g_{Eq} components transformed through geodetic latitude (See Figure 5.2-1):

$$g_{North} = g_{EPA} \cos l - g_{Eq} \sin l \qquad g_{Up} = g_{EPA} \sin l + g_{Eq} \cos l \qquad (11.2.4.3.2.1\text{-}8)$$

where

g_{North} , g_{Up} = Horizontal north and geodetic vertical (up) gravity components.

l = Geodetic latitude from Equations (11.2.4.1.1-1).

From Equation (11.2.4.1.2-1):

$$\sin l = u_{Up_{YI}} \qquad (11.2.4.3.2.1\text{-}9)$$

hence,

$$\cos l = \sqrt{1 - u_{Up_{YI}}^2} \qquad (11.2.4.3.2.1\text{-}10)$$

Substituting in (11.2.4.3.2.1-8) and rearranging to avoid subsequent singularities yields:

$$\left(\frac{g_{North}}{\sqrt{1 - u_{Up_{YI}}^2}}\right) = g_{EPA} - u_{Up_{YI}} \left(\frac{g_{Eq}}{\sqrt{1 - u_{Up_{YI}}^2}}\right)$$

$$g_{Up} = g_{EPA}\, u_{Up_{YI}} + \left(\frac{g_{Eq}}{\sqrt{1 - u_{Up_{YI}}^2}}\right)\left(1 - u_{Up_{YI}}^2\right) \qquad (11.2.4.3.2.1\text{-}11)$$

The desired I Frame components of gravity are the sum of the g_{North} , g_{Up} components along their respective vector directions:

$$\begin{aligned} \underline{g}^I &= g_{Up}\ \underline{u}^I_{Up} + g_{North}\ \underline{u}^I_{North} \\ &= g_{Up}\ \underline{u}^I_{Up} + \left(\frac{g_{North}}{\sqrt{1 - u^2_{Up_{YI}}}}\right)\sqrt{1 - u^2_{Up_{YI}}}\ \underline{u}^I_{North} \end{aligned} \tag{11.2.4.3.2.1-12}$$

where

$\underline{g}^I$ = Gravity vector projected on I Frame coordinates.

Using (11.2.4.2-3) for $\underline{u}^I_{North}$ we then find:

$$\underline{g}^I = g_{Up}\ \underline{u}^I_{Up} + \left(\frac{g_{North}}{\sqrt{1 - u^2_{Up_{YI}}}}\right)\underline{u}^I_{North/Prm} \tag{11.2.4.3.2.1-13}$$

with $\underline{u}^I_{North/Prm}$ defined as:

$$\underline{u}^I_{North/Prm} \equiv \begin{bmatrix} -\,u_{Up_{XI}}\,u_{Up_{YI}} \\ 1 - u^2_{Up_{YI}} \\ -\,u_{Up_{ZI}}\,u_{Up_{YI}} \end{bmatrix} \tag{11.2.4.3.2.1-14}$$

The $u_{Up_{XI}}$, $u_{Up_{YI}}$, $u_{Up_{ZI}}$ terms in (11.2.4.3.2.1-14) are provided from Equations (11.2.4.1.2-1).

11.2.4.3.2.2 Body Frame Integrated Specific Force Acceleration Increments

The integrated specific force acceleration increment is calculated in strapdown sensor or "body" (B Frame) axes (to simulate accelerometer outputs) through a transformation integration algorithm that accounts for B Frame rotation over the update interval. The transformation integration algorithm is based on the assumption that the B Frame specific force acceleration and rotation rate over the update interval is constant.

The transformation integration algorithm is derived by expansion and inversion of the exact equation:

$$\Delta \underline{v}_{SF_m}^{I} = \int_{t_{m-1}}^{t_m} C_B^I \, \underline{a}_{SF}^{B} \, dt \tag{11.2.4.3.2.2-1}$$

where

t_{m-1}, t_m = Time at start and end of the acceleration transformation cycle interval.

C_B^I = Direction cosine matrix that transforms vectors from the B Frame to the I Frame.

$\underline{a}_{SF}^{B}$ = Specific force acceleration in the B Frame (i.e. the vector sensed by the strapdown B Frame mounted accelerometers).

$\Delta \underline{v}_{SF_m}^{I}$ = Integrated specific force in the I Frame over the t_{m-1} to t_m time interval.

The C_B^I matrix in (11.2.4.3.2.2-1) can be equated to its value at t_{m-1} multiplied by a body rotation matrix within the t_{m-1} to t_m interval:

$$C_B^I = C_{B_{m-1}}^{I} \, C_B^{B_{m-1}} \tag{11.2.4.3.2.2-2}$$

where

$C_{B_{m-1}}^{I}$ = Value of C_B^I at the start of the update interval (at t_{m-1}).

$C_B^{B_{m-1}}$ = Direction cosine matrix that transforms vectors from the B Frame at arbitrary time t to the B Frame at $t = t_{m-1}$.

Substituting (11.2.4.3.2.2-2) into (11.2.4.3.2.2-1) then gives:

$$\Delta \underline{v}_{SF_m}^{I} = C_{B_{m-1}}^{I} \, \Delta \underline{v}_{SF_m}^{B_{m-1}} \tag{11.2.4.3.2.2-3}$$

in which we have defined:

$$\Delta \underline{v}_{SF_m}^{B_{m-1}} \equiv \int_{t_{m-1}}^{t_m} C_B^{B_{m-1}} \, \underline{a}_{SF}^{B} \, dt \tag{11.2.4.3.2.2-4}$$

where

$\Delta \underline{v}_{SF_m}^{B_{m-1}}$ = Integrated specific force over the t_{m-1} to t_m time interval calculated in an inertially fixed coordinate frame corresponding to the B Frame attitude at time t_{m-1}.

From Section 7.2.2, Equation (7.2.2-3), we see that $\Delta \underline{v}_{SF_m}^{B_{m-1}}$ is also the input to the Chapter 7 acceleration transformation algorithm. Based on the assumption of constant B Frame acceleration and angular rate, Equations (7.2.2.2.1-5) and (7.2.2.2.1-8) show that (11.2.4.3.2.2-4) is given by:

$$\Delta \underline{v}_{SF_m}^{B_{m-1}} = D_m \, \underline{\upsilon}_m \tag{11.2.4.3.2.2-5}$$

in which:

$$D_m \equiv I + \frac{(1 - \cos \alpha_m)}{\alpha_m^2} \left(\underline{\alpha}_m \times\right) + \frac{1}{\alpha_m^2} \left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) \left(\underline{\alpha}_m \times\right)^2$$

$$\frac{\left(1 - \cos \alpha_m\right)}{\alpha_m^2} = \frac{1}{2\,!} - \frac{\alpha_m^2}{4\,!} + \frac{\alpha_m^4}{6\,!} - \cdots \tag{11.2.4.3.2.2-6}$$

$$\frac{1}{\alpha_m^2} \left(1 - \frac{\sin \alpha_m}{\alpha_m}\right) = \frac{1}{3\,!} - \frac{\alpha_m^2}{5\,!} + \frac{\alpha_m^4}{7\,!} - \cdots$$

where

I = Identity matrix.

$\underline{\upsilon}_m$ = Integrated specific force acceleration in B Frame coordinates over the t_{m-1} to t_m time interval (i.e., the desired Gen Nav simulator integrated accelerometer output increment).

$\underline{\alpha}_m$, α_m = Integrated B Frame inertial angular rate vector and its magnitude over the t_{m-1} to t_m time interval (i.e., the Gen Nav simulator integrated angular rate sensor output increment). From Section 11.2.4.3.1 we see that for the assumed constant angular rate over t_{m-1} to t_m, the $\underline{\alpha}_m$ integrated angular rate vector is the user specified $\underline{\phi}_m$ rotation vector input.

Finally, Equations (11.2.4.3.2.2-3) and (11.2.4.3.2.2-5) are combined and inverted to find for $\underline{\upsilon}_m$:

$$\underline{\upsilon}_m = \left(C_{B_{m-1}}^{I} \, D_m\right)^{-1} \Delta \underline{v}_{SF_m}^{I} \tag{11.2.4.3.2.2-7}$$

The $\Delta \underline{v}_{SF_m}^{I}$ input to (11.2.4.3.2.2-7) is provided from Equation (11.2.4.3.2.1-1).

11.2.4.4 ROLL, PITCH, HEADING ATTITUDE OUTPUTS

Attitude outputs in a strapdown inertial navigation system are typically provided as roll, pitch, heading Euler angles defining the orientation of the strapdown body B Frame relative to local North, East, Down coordinates. The required direction cosine matrix is computed from:

$$C_B^{NED} = \left(C_{NED}^{I}\right)^T C_B^{I} \tag{11.2.4.4-1}$$

where

NED = Local North, East, Down (X, Y, Z) coordinates.

C_B^{NED} = Direction cosine matrix that transforms vectors from the B Frame to the NED Frame.

The C_B^{I} matrix for (11.2.4.4-1) is calculated from the body attitude update Equations (11.2.4.3.1-1). The C_{NED}^{I} matrix is calculated using generalized Equations (3.2.1-6) with the I Frame X, Y axis unit vectors (i.e., $\underline{u}_{East}^{I}$, $\underline{u}_{North}^{I}$) from Equations (11.2.4.2-3) and the I Frame Z axis unit vector (i.e., Down) equal to the negative of $\underline{u}_{Up}^{I}$ from Equations (11.2.4.1.2-1).

$$C_{NED}^{I} = \left[\underline{u}_{East}^{I} \;\; \underline{u}_{North}^{I} \;\; -\underline{u}_{Up}^{I}\right] \tag{11.2.4.4-2}$$

The roll, pitch, heading Euler angles are then extracted from C_B^{NED} as in Equations (11.2.1.3-2) with C_B^{L} interpreted as C_B^{NED}.

12 Strapdown Inertial Navigation Error Equations

12.0 OVERVIEW

In this chapter, we derive time rate differential equations describing the propagation of position, velocity and attitude errors in a strapdown inertial navigation system as a function of input sensor errors and errors in the gravity model utilized in the system software. The chapter begins with a refinement of the basic strapdown inertial navigation equations developed in Chapters 4 and 5 into a format that is more compatible with error equation development. This includes linearization of some of the Chapter 5 equations for earth ellipsoidal surface shape, gravity mass distribution and earth rate centripetal acceleration effects. Three sets of reformatted navigation equations are then developed; a set defined in earth fixed E Frame coordinates, a set defined in the local level navigation N Frame, and a set defined in the inertial non-rotating I Frame. Next, attitude, velocity and position error parameters are defined for each navigation equation set including equivalencies between error parameters, followed by development of the error parameter differential equations in several forms for each of the E, N and I Frame navigation equation sets. A section is included describing the general modeling of inertial sensor error terms appearing in the navigation error parameter differential equations. The chapter concludes with a revision of the error parameter differential equations to facilitate inertial sensor quantization noise modeling and to account for vibration effects.

The coordinate frames used in this chapter are the E, N, L, Geo and B Frames as defined in Section 2.2, and where the I Frame is specialized to:

I Frame = Non-rotating inertial coordinate frame that serves as a stationary reference for defining the angular rotation rates and specific force accelerations relative to inertial space. For this chapter, the Y axis of the I Frame will be specified to lie along the earth's polar rotation axis.

12.1 STRAPDOWN INERTIAL NAVIGATION EQUATIONS

The basic navigation equations implemented in a typical strapdown inertial navigation system are summarized in Table 4.7-1 with earth related parameter equations provided from Table 5.6-1. For error analysis purposes it is advantageous to recognize that for the Table 5.6-1 equations, the earth ellipticity (e) is small (approximately 1/298), the earth mass distribution coefficients (J_2, J_3, etc.) and earth rate contribution to plumb-bob gravity are small (on the order of e) compared to the dominant earth mass attraction term, and the operating altitude for the navigation system is generally small compared to the radius of the earth. As such, first order approximations (linearization) can be applied to parameters containing these terms, as is typically done for many strapdown inertial navigation system software configurations.

This section summarizes the pertinent navigation equation set from Table 4.7-1 and the earth related parameter equation set from Table 5.6-1. Section 12.1.1 then applies the above described linearization process to the Table 5.6-1 equations. Finally, Sections 12.1.2 - 12.1.4 combine the Table 4.7-1 and linearized Table 5.6-1 equation sets to define complete navigation sets in the N, E, and I Frames suitable for error equation development in subsequent sections.

We first summarize the pertinent Table 4.7-1 navigation equations (i.e., Equations (4.1-1), (4.1.1-1), (4.1.1-3), (4.1.1-4), (4.2-1), (4.2-3), (4.1.1-2), (4.4.1.2.1-1), (5.3-18) for $\underline{u}_{ZN}^{N}$, (4.4.1.1-1) (with $\underline{\rho}^{N} = \underline{\omega}_{EN}^{N}$), (4.4.1.2.1-2) and (4.4.1.2.1-3)):

$$\dot{C}_{B}^{L} = C_{B}^{L} \left(\underline{\omega}_{IB}^{B}\times\right) - \left(\underline{\omega}_{IL}^{L}\times\right) C_{B}^{L} \tag{12.1-1}$$

$$\underline{\omega}_{IL}^{L} = C_{N}^{L} \left(\underline{\omega}_{IE}^{N} + \underline{\omega}_{EN}^{N}\right) \tag{12.1-2}$$

$$\underline{\omega}_{IE}^{N} = \left(C_{N}^{E}\right)^{T} \underline{\omega}_{IE}^{E} \tag{12.1-3}$$

$$\underline{\omega}_{IE}^{E} = \left[\ 0 \ \ \omega_e \ \ 0\ \right]^{T} \tag{12.1-4}$$

$$\underline{a}_{SF}^{L} = C_{B}^{L} \underline{a}_{SF}^{B} \tag{12.1-5}$$

$$\underline{a}_{SF}^{N} = C_{L}^{N} \underline{a}_{SF}^{L} \tag{12.1-6}$$

$$C_{N}^{L} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{12.1-7}$$

$$\dot{\underline{v}}^{N} = \underline{a}_{SF}^{N} + \underline{g}_{P}^{N} - \left(\underline{\omega}_{EN}^{N} + 2\,\underline{\omega}_{IE}^{N}\right) \times \underline{v}^{N} - e_{vc_1}\, \underline{u}_{ZN}^{N} \tag{12.1-8}$$

$$\underline{u}_{ZN}^{N} = \left[\ 0 \ \ 0 \ \ 1\ \right]^{T} \tag{12.1-9}$$

$$\dot{C}_{N}^{E} = C_{N}^{E} \left(\underline{\omega}_{EN}^{N}\times\right) \tag{12.1-10}$$

$$\dot{h} = \underline{v}^{N} \cdot \underline{u}_{ZN}^{N} - e_{vc_2} \tag{12.1-11}$$

$$\partial h = h - h_{Prsr}$$
$$e_{vc_1} = e_{vc_3} + C_2\,\partial h \qquad e_{vc_2} = C_3\,\partial h \qquad \dot{e}_{vc_3} = C_1\,\partial h \tag{12.1-12}$$

where

C_B^L, C_L^N, C_N^E = Direction cosine matrices that transform vectors from the B to L, L to N and N to E Frames.

$\underline{\omega}_{IB}^B$ = Angular rate of the B Frame relative to the I Frame as projected on B Frame axes (i.e., the angular rate sensed by strapdown angular rate sensors).

$\underline{\omega}_{IL}^L$ = Angular rate of the L Frame relative to the I Frame as projected on L Frame axes.

$\underline{\omega}_{IE}^N$ = Angular rate of the E Frame relative to the I Frame (i.e., earth's angular rate relative to inertial space) as projected on N Frame axes.

ω_e = Magnitude of earth's angular rate relative to inertial space.

$\underline{\omega}_{EN}^N$ = Angular rate of the N Frame relative to the E Frame ("transport rate") as projected on N Frame axes (identified as $\underline{\rho}^N$ in Equation (4.4.1.1-1)).

$\underline{a}_{SF}^B$ = Specific force acceleration relative to inertial space as projected on B Frame axes (i.e., the acceleration sensed by strapdown accelerometers which is total acceleration exclusive of gravitational effects).

$\underline{a}_{SF}^L, \underline{a}_{SF}^N$ = Specific force acceleration projected on L and N Frame axes.

$\underline{v}^N$ = Velocity of the navigation system relative to the earth as projected on N Frame axes.

$\underline{g}_P^N$ = Plumb-bob gravity as projected on N Frame axes.

$\underline{u}_{ZN}^N$ = Unit vector along the N Frame Z axis as projected on N Frame axes. From the Section 2.2 definition of the N and L Frames, this vector lies along the local geodetic upward vertical and the L Frame negative Z axis.

h = Altitude of the navigation system above the earth reference geoid surface.

h_{Prsr} = Altitude as determined from a pressure transducer.

∂h = The difference between h and h_{Prsr} used to control inertial navigation system vertical channel error.

$e_{vc_1}, e_{vc_2}, e_{vc_3}$ = Vertical channel control signals.

C_1, C_2, C_3 = Vertical channel control gains.

We note for later use that the formal definition for $\underline{v}$ is given by Equation (4.3-1) in the E Frame. Rearrangement of this equation and adding the Equation (4.4.1.2.1-2) vertical loop control signal in the vertical direction obtains:

$$\dot{\underline{R}}^E = \underline{v}^E - e_{vc_2}\, \underline{u}_{ZN}^E \tag{12.1-13}$$

where

$\underline{R}^E$ = Position vector from earth's center to the navigation system as projected on E Frame axes.

$\underline{u}_{ZN}^E$ = Unit vector along the N Frame Z axis (local upward geodetic vertical) as projected on E Frame axes.

We also note from Equation (4.3-16) with the (4.4.1.2.1-2) vertical channel control signal, that the equivalent to Equation (12.1-8) in the E Frame is:

$$\dot{\underline{v}}^E = \underline{a}_{SF}^E + \underline{g}_P^E - 2\, \underline{\omega}_{IE}^E \times \underline{v}^E - e_{vc_1}\, \underline{u}_{ZN}^E \tag{12.1-14}$$

Next, we summarize the pertinent Table 5.6-1 earth related parameter equations with some minor rearrangements (i.e., (5.3-16), (5.1-10), (5.2.1-4), (5.2.1-5), (5.2.4-25), (5.2.4-37), (5.3-18), (5.3-17) with $\underline{\rho}^N = \underline{\omega}_{EN}^N$, (5.2.2-3), (5.2.2-4), (5.4-1), (5.4-2), (5.2.3-5), (5.4-3) and (5.4.1-9)), and in which $\underline{u}_{ZN}$ is substituted for $\underline{u}_{Up}$ (the unit vector upward along the local vertical). These equations will be modified in Section 12.1.1 for the linearization approximation process.

$$u_{ZN_{YE}} = D_{23} \tag{12.1-15}$$

$$R_S' = R_0 \Big/ \sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \tag{12.1-16}$$

$$R_S = R_S' \sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^4 - 1 \right]} \tag{12.1-17}$$

$$R^2 = R_S^2 + 2\, h\, R_0^2 / R_S' + h^2 \tag{12.1-18}$$

$$r_{ls} = \frac{R_S'^3}{R_0^2} (1 - e)^2 \tag{12.1-19}$$

$$r_l = r_{ls} + h \tag{12.1-20}$$

$$F_C^N = \begin{bmatrix} F_{C_{11}} & F_{C_{12}} & 0 \\ F_{C_{21}} & F_{C_{22}} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$F_{C_{11}} = \frac{1}{r_l}\left(1 + D_{21}^2 f_{eh}\right) \qquad F_{C_{12}} = \frac{1}{r_l} D_{21} D_{22} f_{eh} \tag{12.1-21}$$

$$F_{C_{21}} = \frac{1}{r_l} D_{21} D_{22} f_{eh} \qquad F_{C_{22}} = \frac{1}{r_l}\left(1 + D_{22}^2 f_{eh}\right)$$

$$f_e = \frac{(1 - e)^2 - 1}{1 + D_{23}^2\left[(1 - e)^2 - 1\right]} \qquad f_h = \frac{1}{1 + h / R_S'} \qquad f_{eh} = f_e f_h$$

$$\underline{\omega}_{EN}^N = F_C^N \left(\underline{u}_{ZN}^N \times \underline{v}^N\right) + \rho_{ZN}\, \underline{u}_{ZN}^N \tag{12.1-22}$$

$$\cos \phi = u_{ZN_{YE}} \left[(1 - e)^2 R_S' + h\right] / R \tag{12.1-23}$$

$$\sin \phi = \sqrt{1 - u_{ZN_{YE}}^2}\left(R_S' + h\right) / R \tag{12.1-24}$$

For $h \geq 0$:

$$g_r = -\frac{\mu}{R^2}\left[1 - \frac{3}{2} J_2 \left(\frac{R_0}{R}\right)^2 \left(3 \cos^2 \phi - 1\right) - 2 J_3 \left(\frac{R_0}{R}\right)^3 \cos \phi \left(5 \cos^2 \phi - 3\right) - \cdots\right]$$

$$\left(\frac{g_\phi}{\sin \phi}\right) = 3 \frac{\mu}{R^2}\left(\frac{R_0}{R}\right)^2 \left[J_2 \cos \phi + \frac{1}{2} J_3 \frac{R_0}{R}\left(5 \cos^2 \phi - 1\right) + \cdots\right] \tag{12.1-25}$$

For $h < 0$:

$$g_r = \frac{R}{R_S} g_{r_S} \qquad \left(\frac{g_\phi}{\sin \phi}\right) = \frac{R}{R_S}\left(\frac{g_\phi}{\sin \phi}\right)_S$$

$$\sin \partial l = u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2}\left[1 - (1 - e)^2\right] R_S' / R$$

$$\cos \partial l = \left\{\left(1 - u_{ZN_{YE}}^2\left[1 - (1 - e)^2\right]\right) R_S' + h\right\} / R \tag{12.1-26}$$

$$g_{Up} = g_r \cos \partial l - g_\phi \sin \partial l$$
$$g_{North} = - g_\phi \cos \partial l - g_r \sin \partial l \tag{12.1-27}$$

$$g_{P_{North}} = g_{North} - \left(R_S^{'} + h\right) \omega_e^2 \, u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2}$$
$$g_{P_{Up}} = g_{Up} + \left(R_S^{'} + h\right) \omega_e^2 \left(1 - u_{ZN_{YE}}^2\right) \tag{12.1-28}$$

where

$u_{ZN_{YE}}$ = Y axis component of $\underline{u}_{ZN}^E$.

D_{2j} = Element in row 2, column j of C_N^E.

R_0 = Earth's equatorial radius.

e = Earth's ellipticity.

R_S = Distance from earth's center to the point on the earth's surface directly below (above) the navigation system.

$R_S^{'}$ = Modified R_S parameter.

R = Distance from earth's center to the navigation system.

r_{l_S} = Radius of curvature of the earth's surface in the latitude direction for a point on the earth's surface directly below (above) the navigation system.

r_l = Equivalent to the r_{l_S} radius of curvature, but at the navigation system altitude.

F_C^N = Curvature matrix in the N Frame.

f_e = Function of e.

f_h = Function of h (and $R_S^{'}$).

f_{eh} = Product of f_e and f_h.

ρ_{ZN} = N Frame Z axis component of transport rate $\underline{\omega}_{EN}^N$ (See Section 4.5 for options).

ϕ = Angle from earth's positive rotation ("polar") axis to the line from earth's center to the navigation system (See Figure 5.2-1).

∂l = Angle from "geodetic vertical" (locally up, directly below (above) the navigation system and perpendicular to the surface of the earth) to the line from earth's center to the navigation system (See Figure 5.2-1).

g_r = Earth's mass attraction gravity component along a line from the center of the earth to the navigation system measured positive away from earth's center.

g_ϕ = Earth's mass attraction gravity component in the plane of earth's polar axis and the line from earth's center to the navigation system, and perpendicular to the line from earth's center to the navigation system (measured positive pointing away from the earth's positive polar axis).

$g_{r_S}, \left(\dfrac{g_\phi}{\sin\phi}\right)_S$ = Values for g_r and $\left(\dfrac{g_\phi}{\sin\phi}\right)$ calculated using the first set in Equations (12.1-25) (for $h \geq 0$) with R set to R_S.

g_{Up} = Earth's mass attraction gravity component along the local geodetic vertical (i.e., along $\underline{u}_{ZN}$ which lies along the N Frame Z axis).

g_{North} = Earth's mass attraction gravity component along the local horizontal north direction.(i.e., along the Geo Frame Y axis).

$g_{P_{North}}, g_{P_{Up}}$ = Plumb-bob gravity components along the local horizontal North and geodetic vertical Up directions (See Figure 5.2-1).

Equation (12.1-18) can be revised by substituting (12.1-16) and (12.1-17):

$$
\begin{aligned}
R^2 &= R_S^2 + 2\,h\,R_0^2 / R_S' + h^2 = R_S^2 + 2\,h\,\frac{R_0^2}{R_S'^{\,2}}\,\frac{R_S'}{R_S}\,R_S + h^2 \\
&= R_S^2 + 2\,h\,R_S\,\frac{\left(1 + u_{ZN_{YE}}^2\left[(1-e)^2 - 1\right]\right)}{\sqrt{1 + u_{ZN_{YE}}^2\left[(1-e)^4 - 1\right]}} + h^2 \qquad (12.1\text{-}29) \\
&= R_S^2 + 2\,h\,R_S + h^2 + 2\,h\,R_S\left(\frac{\left(1 + u_{ZN_{YE}}^2\left[(1-e)^2 - 1\right]\right)}{\sqrt{1 + u_{ZN_{YE}}^2\left[(1-e)^4 - 1\right]}} - 1\right)
\end{aligned}
$$

or

$$
R^2 = \left(R_S + h\right)^2 + 2\,h\,R_S\left(\frac{\left(1 + u_{ZN_{YE}}^2\left[(1-e)^2 - 1\right]\right)}{\sqrt{1 + u_{ZN_{YE}}^2\left[(1-e)^4 - 1\right]}} - 1\right) \qquad (12.1\text{-}30)
$$

A useful vector relation for $\underline{u}_{ZN}^E$ can be derived from Equation (5.2.2-1) repeated below:

$$
\begin{aligned}
R_{XE} &= u_{ZN_{XE}} \left(R_S' + h \right) \\
R_{YE} &= u_{ZN_{YE}} \left[(1 - e)^2 R_S' + h \right] \\
R_{ZE} &= u_{ZN_{ZE}} \left(R_S' + h \right)
\end{aligned}
\tag{12.1-31}
$$

In vector form, Equations (12.1-31) are:

$$
\underline{R}^E = \left(R_S' + h \right) \underline{u}_{ZN}^E + \left[(1 - e)^2 - 1 \right] u_{ZN_{YE}} R_S' \, \underline{u}_{YE}^E \tag{12.1-32}
$$

with from the definition of the E Frame:

$$
\underline{u}_{YE}^E = \left[\begin{array}{ccc} 0 & 1 & 0 \end{array} \right]^T \tag{12.1-33}
$$

where

$\underline{u}_{YE}^E$ = Unit vector along the E Frame Y axis (i.e., along earth's polar rotation axis).

The converse of (12.1-32) is:

$$
\underline{u}_{ZN}^E = \frac{1}{\left(R_S' + h \right)} \underline{R}^E - \left[(1 - e)^2 - 1 \right] u_{ZN_{YE}} \frac{R_S'}{\left(R_S' + h \right)} \underline{u}_{YE}^E \tag{12.1-34}
$$

The classical definition of $\underline{u}_{ZN}^E$ will also be useful:

$$
\underline{u}_{ZN}^E = C_N^E \, \underline{u}_{ZN}^N \tag{12.1-35}
$$

Finally, we will make use of a unit vector in the horizontal north direction defined in E Frame coordinates by Equations (5.4.1-6) as:

$$
\underline{u}_{North}^E = \frac{1}{\sqrt{1 - u_{ZN_{YE}}^2}} \begin{bmatrix} - u_{ZN_{XE}} u_{ZN_{YE}} \\ 1 - u_{ZN_{YE}}^2 \\ - u_{ZN_{ZE}} u_{ZN_{YE}} \end{bmatrix} \tag{12.1-36}
$$

where

$\underline{u}_{North}^E$ = Unit vector along the local horizontal north direction as projected on E Frame axes.

$u_{ZN_{XE}}, u_{ZN_{YE}}, u_{ZN_{ZE}}$ = X, Y, Z components of $\underline{u}_{ZN}^E$.

12.1.1 APPLICATION OF THE LINEARIZATION PROCESS

We now apply the linearization process to the e terms in Equations (12.1-16), (12.1-17), (12.1-19), (12.1-21) with (12.1-5) for D_{23}, (12.1-26), (12.1-30) and (12.1-34). Successive application of the binomial theorem to these terms while dropping terms of order e^2 and higher obtains :

$$
\begin{aligned}
&1 / \sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \approx 1 / \sqrt{1 - 2\, u_{ZN_{YE}}^2\, e} \approx 1 + u_{ZN_{YE}}^2\, e \\
&\sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^4 - 1 \right]} \approx \sqrt{1 - 4\, u_{ZN_{YE}}^2\, e} \approx 1 - 2\, u_{ZN_{YE}}^2\, e \\
&(1 - e)^2 \approx 1 - 2\, e \qquad\qquad (12.1.1\text{-}1) \\
&\frac{(1 - e)^2 - 1}{1 + u_{ZN_{YE}}^2 \left[(1 - e)^2 - 1 \right]} \approx - 2\, e \\
&\left(\frac{\left(1 + u_{ZN_{YE}}^2 \left[(1 - e)^2 - 1 \right] \right)}{\sqrt{1 + u_{ZN_{YE}}^2 \left[(1 - e)^4 - 1 \right]}} - 1 \right) \approx 0
\end{aligned}
$$

Applying Equations (12.1.1-1) and the binomial theorem to Equations (12.1-16), (12.1-17), (12.1-19), (12.1-21), (12.1-26), (12.1-30) and terms containing $R_S' + h$ in (12.1-34), then obtains upon appropriate combination and rearrangement:

$$
\begin{aligned}
&R_S' \approx \left(1 + u_{ZN_{YE}}^2\, e \right) R_0 \\
&R_S \approx \left(1 - 2\, u_{ZN_{YE}}^2\, e \right) R_S' \approx \left(1 - u_{ZN_{YE}}^2\, e \right) R_0 \\
&r_{ls} = \frac{R_S'^{\,3}}{R_0^2} (1 - e)^2 = \frac{R_S'^{\,2}}{R_0^2} \frac{R_S'}{R_S} R_S (1 - e)^2 \qquad\qquad (12.1.1\text{-}2) \\
&\quad \approx \frac{\left(1 + 2\, u_{ZN_{YE}}^2\, e \right)}{\left(1 - 2\, u_{ZN_{YE}}^2\, e \right)} (1 - 2\, e) R_S \approx \left(1 + 4\, u_{ZN_{YE}}^2\, e - 2\, e \right) R_S \\
&\quad = \left[1 + 2 \left(2\, u_{ZN_{YE}}^2 - 1 \right) e \right] R_S
\end{aligned}
$$

(Continued)

$$f_{eh} \approx - \frac{2\,e}{1 + h / R_S^{'}}$$

$$\sin \partial l \approx \partial l \approx 2\,e\,u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2}\; R_S^{'} / R$$

$$\cos \partial l = \sqrt{1 - \sin^2 \partial l} \approx 1$$

$$R \approx R_S + h \approx R_0 + h - e\,u_{ZN_{YE}}^2 R_0 \qquad \text{(12.1.1-2) (Continued)}$$

$$R_S^{'} + h = R_S + h + R_S^{'} - R_S \approx R + 2\,e\,u_{ZN_{YE}}^2 R_0$$

$$\frac{1}{R_S^{'} + h} \approx \frac{1}{R + 2\,e\,u_{ZN_{YE}}^2 R_0} = \frac{1}{R\left(1 + 2\,e\,u_{ZN_{YE}}^2 \dfrac{R_0}{R}\right)} \approx \frac{1}{R}\left(1 - 2\,e\,u_{ZN_{YE}}^2 \frac{R_0}{R}\right)$$

Recall from Equation (5.2.3-1) that $u_{ZN_{YE}}$ equals the sine of geodetic latitude. Note that at 45 degrees latitude (an average operating latitude for many systems) the bracketed term in the Equations (12.1.1-2) r_{l_S} expression is, therefore, equal to zero. Thus, the average value for r_{l_S} is R_S, which intuitively seems correct.

The gravity equations can be linearized by recognizing that the earth mass distribution coefficients (J_2, J_3, etc.) are on the order of e or smaller. Then Equations (12.1-25) show that g_ϕ is much smaller than g_r (by an order of e factor). Thus, with sin ∂l and cos ∂l from (12.1.1-2), Equations (12.1-28) with (12.1-27) simplify to:

$$\begin{aligned} g_{P_{North}} &= - g_\phi - g_r\, 2\,e\,u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2}\; R_S^{'} / R \\ &\quad - u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left(R_S^{'} + h\right) \omega_e^2 \\ &= - g_\phi - u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left[2\,e\,g_r\,R_S^{'} / R + \left(R_S^{'} + h\right) \omega_e^2\right] \end{aligned} \qquad \text{(12.1.1-3)}$$

$$g_{P_{Up}} = g_r + \left(1 - u_{ZN_{YE}}^2\right)\left(R_S^{'} + h\right) \omega_e^2 \qquad \text{(12.1.1-4)}$$

The $g_{P_{North}}$ expression can be further reduced by dropping mass distribution terms of J_3 and higher in g_ϕ and dropping all mass distribution terms in g_r as second order. We also note that use of the $R_S^{'} + h$ approximation from (12.1.1-2) in Equations (12.1-23) and (12.1-24) shows that to zero order in e (i.e., dropping all power of e terms):

$$\cos\phi \approx u_{ZN_{YE}}\left[(1 - 2\,e)\,R_S' + h\right] / R \;=\; u_{ZN_{YE}}\left[R_S' + h - 2\,e\,R_S'\right] / R$$
$$\approx u_{ZN_{YE}}\left[R + 2\,e\,u_{ZN_{YE}}^2\,R_0 - 2\,e\,R_S'\right] / R \;\approx\; u_{ZN_{YE}} \qquad (12.1.1\text{-}5)$$
$$\sin\phi \approx \sqrt{1 - u_{ZN_{YE}}^2}\left(R + 2\,e\,u_{ZN_{YE}}^2\,R_0\right) / R \;\approx\; \sqrt{1 - u_{ZN_{YE}}^2}$$

With (12.1-25) and (12.1.1-5), Equation (12.1.1-3) becomes for positive altitude h:

For $h \geq 0$:

$$g_{P_{North}} \approx -3\,\frac{\mu}{R^2}\left(\frac{R_0}{R}\right)^2 J_2 \sin\phi\cos\phi$$
$$- u_{ZN_{YE}}\sqrt{1 - u_{ZN_{YE}}^2}\left(-2\,e\,\frac{\mu}{R^2}\,\frac{R_S'}{R} + \left(R_S' + h\right)\omega_e^2\right) \qquad (12.1.1\text{-}6)$$
$$\approx u_{ZN_{YE}}\sqrt{1 - u_{ZN_{YE}}^2}\left\{\frac{\mu}{R^2}\left[-3\left(\frac{R_0}{R}\right)^2 J_2 + 2\,e\,R_S' / R\right] - \left(R_S' + h\right)\omega_e^2\right\}$$

It is readily verified by numerical substitution that for h on the order of R_S' (or smaller) the $\left(R_S' + h\right)\omega_e^2$ term in (12.1.1-6) is on the order of the terms on the left. Thus, with the $R_S' + h$, R, R_S and R_S' approximations from (12.1.1-2), Equation (12.1.1-6) to first order in e further reduces to:

For $h \geq 0$:

$$g_{P_{North}} \approx u_{ZN_{YE}}\sqrt{1 - u_{ZN_{YE}}^2}\left\{\frac{\mu}{R^2}\left[-3\left(\frac{R_0}{R}\right)^2 J_2 + 2\,e\,R_0 / R\right] - R\,\omega_e^2\right\}$$
$$\approx u_{ZN_{YE}}\sqrt{1 - u_{ZN_{YE}}^2}\left\{\frac{\mu\,R_0}{(R_0 + h)^3}\left[-3\,\frac{R_0}{(R_0 + h)}\,J_2 + 2\,e\right] - (R_0 + h)\,\omega_e^2\right\} \qquad (12.1.1\text{-}7)$$
$$= u_{ZN_{YE}}\sqrt{1 - u_{ZN_{YE}}^2}\left\{\frac{\mu}{R_0^2}\,\frac{1}{\left(1 + \frac{h}{R_0}\right)^3}\left[-\frac{3}{\left(1 + \frac{h}{R_0}\right)}\,J_2 + 2\,e\right] - R_0\left(1 + \frac{h}{R_0}\right)\omega_e^2\right\}$$

We now make the approximation that h will be small compared to R_0 which is valid for position locations in the earth's atmosphere and for low altitude earth satellites. Then to first order in $\frac{h}{R_0}$, Equation (12.1.1-7) simplifies to:

For $h \geq 0$:

$$g_{P_{North}} \approx u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left\{ \frac{\mu}{R_0^2} \left(1 - 3 \frac{h}{R_0}\right) \left[- 3 \left(1 - \frac{h}{R_0}\right) J_2 + 2 \, e \right] - R_0 \left(1 + \frac{h}{R_0}\right) \omega_e^2 \right\}$$

$$\approx u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left\{ \left[\frac{\mu}{R_0^2} \left(2 \, e - 3 \, J_2\right) - R_0 \, \omega_e^2 \right] + \left[6 \frac{\mu}{R_0^3} \left(2 \, J_2 - e\right) - \omega_e^2 \right] h \right\} \qquad (12.1.1\text{-}8)$$

An interesting fact of nature is that the horizontal plumb-bob gravity component on the surface of the earth is almost zero (within approximately 20 micro-g's). Scientists use this fact as evidence of the earth's surface floating on a molten core at the ω_e spin rate. A spinning molten mass that contains itself by its own gravity will reach a steady state condition in which the horizontal specific force acceleration at the surface (i.e., the acceleration that will move the surface fluid mass toward equilibrium) is zero. From Equation (12.1.1-8), we see that for $g_{P_{North}}$ to be zero on the earth's surface (i.e., $h = 0$), the $\frac{\mu}{R_0^2} \left(2 \, e - 3 \, J_2\right) - R_0 \, \omega_e^2$ term must be zero. Using the values for μ, e, J_2, R_0 and ω_e given in Table 5.6-1 of Chapter 5, $\frac{\mu}{R_0^2} \left(2 \, e - 3 \, J_2\right) - R_0 \, \omega_e^2$ computes to be - 1.174 E-4 ft/sec^2 or - 3.65 micro-g). Thus, Equation (12.1.1-8) reduces to the simplified approximate form:

For $h \geq 0$:

$$g_{P_{North}} \approx u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left[6 \frac{\mu}{R_0^3} \left(2 \, J_2 - e\right) - \omega_e^2 \right] h \qquad (12.1.1\text{-}9)$$

Using(12.1.1-2) for $R_S' + h$, (12.1.1-5) for $\cos \phi$, and g_r from (12.1-25) in Equation (12.1.1-4), we also obtain for $g_{P_{Up}}$ when altitude h is positive:

For $h \geq 0$:

$$g_{P_{Up}} \approx - \frac{\mu}{R^2} \left[1 - \frac{3}{2} J_2 \left(\frac{R_0}{R}\right)^2 \left(3 \cos^2 \phi - 1\right) \right] + \left(1 - u_{ZN_{YE}}^2\right) \left(R_S' + h\right) \omega_e^2$$

$$= - \frac{\mu}{R^2} + \frac{3}{2} J_2 \, \frac{\mu}{R^2} \left(\frac{R_0}{R}\right)^2 \left(3 \, u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) \left(R_S' + h\right) \omega_e^2 \qquad (12.1.1\text{-}10)$$

$$\approx - \frac{\mu}{R^2} + \frac{3}{2} J_2 \, \frac{\mu}{R^2} \left(\frac{R_0}{R}\right)^2 \left(3 \, u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R \, \omega_e^2$$

For plumb-bob gravity at negative altitudes, we apply the same approximation used in formulating (5.4-2) for $h < 0$ (duplicated earlier in (12.1-25)), namely that the mass attraction gravity components will be linearly proportional to the radial distance from earth's center (R) and will equal the values of the (12.1-25) $h \geq 0$ gravity components at the earth geoid surface (i.e., at h = 0). If we look at Equations (12.1-27) and (12.1-28), we see that plumb-bob gravity is formed as a linear combination of the mass attraction components (g_r and g_ϕ) plus a centripetal term proportional to $\left(R_S' + h\right) \omega_e^2$ (or approximately $R\ \omega_e^2$; i.e., proportional to R). We can conclude, therefore, that the plumb-bob gravity components can also be approximated for negative altitudes as being linearly proportional to the distance from earth's center, matching the Equation (12.1.1-9) and (12.1.1-10) values for h = 0. Then Equations (12.1.1-9) and (12.1.1-10) with the previous stated negative altitude approximation for the plumb-bob gravity components form the following vector equivalent in the N Frame:

$$\underline{g}_P^N = - H(R)\ \underline{u}_{ZN}^N + \partial g_{P_{Up}}\ \underline{u}_{ZN}^N + \partial g_{P_{North}}\ C_E^N\ \underline{u}_{North}^E \qquad (12.1.1\text{-}11)$$

with

For $h \geq 0$:

$$H(R) = \frac{\mu}{R^2}$$

$$\partial g_{P_{Up}} \approx \frac{3}{2} J_2 \frac{\mu}{R^2} \left(\frac{R_0}{R}\right)^2 \left(3\ u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R\ \omega_e^2$$

$$\partial g_{P_{North}} \approx u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left[6 \frac{\mu}{R_0^3} \left(2 J_2 - e\right) - \omega_e^2\right] h \qquad (12.1.1\text{-}12)$$

For $h < 0$:

$$H(R) = \frac{R}{R_S} \frac{\mu}{R_S^2}$$

$$\partial g_{P_{Up}} \approx \frac{R}{R_S} \left[\frac{3}{2} J_2 \frac{\mu}{R_S^2} \left(\frac{R_0}{R_S}\right)^2 \left(3\ u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R_S\ \omega_e^2\right]$$

$$\partial g_{P_{North}} \approx 0$$

where

$H(R)$ = Gravity magnitude parameter that characterizes the fundamental difference between gravity values above and below the earth's surface.

$\partial g_{P_{Up}}$ = Small variation in the vertical component of plumb-bob gravity from the nominal spherical earth uniform density approximation (i.e., $- H(R)$) due to earth's mass distribution and earth's rotation centripetal acceleration effects.

$\partial g_{P_{North}}$ = Small variation in the north component of plumb-bob gravity from the nominal spherical earth uniform density approximation (i.e., zero) due to earth's mass distribution and earth's rotation centripetal acceleration effects.

Finally, the approximate first order equivalent to Equations (12.1-16) - (12.1-28) and (12.1-30) (i.e., first order in e, $\frac{h}{R_0}$ and earth mass distribution coefficients for selective terms) is provided by incorporating Equations (12.1.1-2) and (12.1.1-11) - (12.1.1-12) with some minor refinements:

$$R_S' \approx \left(1 + u_{ZN_{YE}}^2\, e\right) R_0$$

$$R_S \approx \left(1 - 2\, u_{ZN_{YE}}^2\, e\right) R_S' \approx \left(1 - u_{ZN_{YE}}^2\, e\right) R_0$$

$$R \approx R_S + h$$

$$r_{ls} = \left[1 + 2\left(2\, u_{ZN_{YE}}^2 - 1\right) e\right] R_S$$

$$r_l = r_{ls} + h$$

$$\partial G_C^N = -\frac{2\,e}{\left(1 + h / R_S'\right)} \begin{bmatrix} D_{21}^2 & D_{21}\, D_{22} & 0 \\ D_{21}\, D_{22} & D_{22}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad (12.1.1\text{-}13)$$

$$\underline{\omega}_{EN}^N = \rho_{ZN}\, \underline{u}_{ZN}^N + \frac{1}{r_l}\left(\underline{u}_{ZN}^N \times \underline{v}^N\right) + \frac{1}{r_l}\, \partial G_C^N \left(\underline{u}_{ZN}^N \times \underline{v}^N\right)$$

$$\underline{g}_P^N = -\,H(R)\ \underline{u}_{ZN}^N + \partial g_{P_{Up}}\, \underline{u}_{ZN}^N + \partial g_{P_{North}}\, C_E^N\, \underline{u}_{North}^E$$

For $h \geq 0$:

$$H(R) = \frac{\mu}{R^2}$$

$$\partial g_{P_{Up}} \approx \frac{3}{2}\, J_2\, \frac{\mu}{R^2} \left(\frac{R_0}{R}\right)^2 \left(3\, u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R\, \omega_e^2$$

$$\partial g_{P_{North}} \approx u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left[6\, \frac{\mu}{R_0^{\,3}} \left(2\, J_2 - e\right) - \omega_e^2\right] h$$

(Continued)

For $h < 0$:

$$H(R) = \frac{R}{R_S} \frac{\mu}{R_S^2} \qquad (12.1.1\text{-}13) \text{ (Continued)}$$

$$\partial g_{P_{Up}} \approx \frac{R}{R_S} \left[\frac{3}{2} J_2 \frac{\mu}{R_S^2} \left(\frac{R_0}{R_S} \right)^2 \left(3\, u_{ZN_{YE}}^2 - 1 \right) + \left(1 - u_{ZN_{YE}}^2 \right) R_S\, \omega_e^2 \right]$$

$$\partial g_{P_{North}} \approx 0$$

where

∂G_C^N = Portion of $r_l\, F_C^N$ that accounts for earth's oblateness.

Additionally, the Equation (12.1-34) $\underline{u}_{ZN}^E$ expression is linearized to first order in e using the $\frac{1}{\left(R_S' + h\right)}$, R_S', R_S and R approximations from (12.1.1-2):

$$\begin{aligned} \underline{u}_{ZN}^E &= \frac{1}{\left(R_S' + h\right)} \underline{R}^E - \left[(1 - e)^2 - 1 \right] u_{ZN_{YE}} \frac{R_S'}{\left(R_S' + h\right)} \underline{u}_{YE}^E \\ &\approx \frac{1}{R} \left(1 - 2\, e\, u_{ZN_{YE}}^2 \frac{R_0}{R} \right) \underline{R}^E + 2\, e\, u_{ZN_{YE}} \frac{R_0}{R} \underline{u}_{YE}^E \qquad (12.1.1\text{-}14) \\ &= \frac{1}{R} \underline{R}^E - 2\, e\, u_{ZN_{YE}} \frac{R_0}{R} \left(u_{ZN_{YE}} \frac{1}{R} \underline{R}^E - \underline{u}_{YE}^E \right) \end{aligned}$$

We will make use of (12.1.1-13) - (12.1.1-14) in subsequent sections.

12.1.2 NAVIGATION EQUATIONS FOR N FRAME ERROR ANALYSIS

For error analysis purposes we will analyze three sets of navigation equations; a set in which errors are defined in the navigation N Frame, a set in which errors are defined in the earth E Frame and a set in which errors are defined in the inertial I Frame. This section develops the N Frame navigation equation set.

We begin by first eliminating the L Frame from the Section 12.1 basic navigation equations. Recognizing from Equation (12.1-7) that C_L^N is constant and applying the Equation (3.1.1-39) similarity transformation rule for cross-product operators yields from Equation (12.1-1):

$$
\begin{aligned}
C_L^N \dot{C}_B^L &= \frac{d}{dt}\left(C_L^N C_B^L\right) = \dot{C}_B^N \\
&= C_L^N C_B^L \left(\underline{\omega}_{IB}^B \times\right) - C_L^N \left(\underline{\omega}_{IL}^L \times\right) C_B^L \qquad (12.1.2\text{-}1) \\
&= C_L^N C_B^L \left(\underline{\omega}_{IB}^B \times\right) - C_L^N \left(\underline{\omega}_{IL}^L \times\right) \left(C_L^N\right)^T C_L^N C_B^L \\
&= C_B^N \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IL}^N \times\right) C_B^N
\end{aligned}
$$

Since the L Frame is fixed relative to the N Frame we can write:

$$
\underline{\omega}_{IL} = \underline{\omega}_{IN} \qquad (12.1.2\text{-}2)
$$

where

$\underline{\omega}_{IN}$ = Angular rate of the N Frame relative to the inertial non-rotating I Frame.

With (12.1.2-2), Equation (12.1.2-1) becomes:

$$
\dot{C}_B^N = C_B^N \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IN}^N \times\right) C_B^N \qquad (12.1.2\text{-}3)
$$

Transforming Equation (12.1-2) to the N Frame obtains:

$$
\underline{\omega}_{IN}^N = \underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N \qquad (12.1.2\text{-}4)
$$

Combining Equations (12.1-5) - (12.1-6) and substitution into (12.1-8) yields:

$$
\dot{\underline{v}}^N = C_B^N \underline{a}_{SF}^B + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N\right) \times \underline{v}^N - e_{vc_1} \underline{u}_{ZN}^N \qquad (12.1.2\text{-}5)
$$

Substituting Equations (12.1.2-3) - (12.1.2-5) for the equivalent terms in Equations (12.1-1) - (12.1-12) and retention of applicable terms from Equations (12.1-1) - (12.1-12), (12.1.1-13), and (12.1-35) - (12.1-36) then provides the desired N Frame navigation equation set for subsequent error analysis:

$$
\begin{aligned}
\dot{C}_B^N &= C_B^N \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IN}^N \times\right) C_B^N \\
\underline{\omega}_{IN}^N &= \underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N \qquad (12.1.2\text{-}6) \\
\underline{\omega}_{IE}^N &= \left(C_N^E\right)^T \underline{\omega}_{IE}^E
\end{aligned}
$$

(Continued)

$$\underline{\omega}_{IE}^{E} = \left[\; 0 \;\; \omega_e \;\; 0 \;\right]^T$$

$$\dot{\underline{v}}^N = C_B^N \, \underline{a}_{SF}^B + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \right) \times \underline{v}^N - e_{vc_1} \, \underline{u}_{ZN}^N$$

$$\underline{u}_{ZN}^N = \left[\; 0 \;\; 0 \;\; 1 \;\right]^T$$

$$\dot{C}_N^E = C_N^E \left(\underline{\omega}_{EN}^N \times \right)$$

$$\dot{h} = \underline{v}^N \cdot \underline{u}_{ZN}^N - e_{vc_2}$$

$$\partial h = h - h_{Prsr}$$

$$e_{vc_1} = e_{vc_3} + C_2 \, \partial h \qquad e_{vc_2} = C_3 \, \partial h \qquad \dot{e}_{vc_3} = C_1 \, \partial h$$

$$R_S' \approx \left(1 + u_{ZN_{YE}}^2 \, e \right) R_0$$

$$R_S \approx \left(1 - u_{ZN_{YE}}^2 \, e \right) R_0 \qquad \text{(12.1.2-6) (Continued)}$$

$$R \approx R_S + h$$

$$r_{l_S} = \left[1 + 2 \left(2 \, u_{ZN_{YE}}^2 - 1 \right) e \right] R_S$$

$$r_l = r_{l_S} + h$$

$$\partial G_C^N = - \frac{2\,e}{\left(1 + h / R_S' \right)} \begin{bmatrix} D_{21}^2 & D_{21} \, D_{22} & 0 \\ D_{21} \, D_{22} & D_{22}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\underline{\omega}_{EN}^N = \rho_{ZN} \, \underline{u}_{ZN}^N + \frac{1}{r_l} \left(\underline{u}_{ZN}^N \times \underline{v}^N \right) + \frac{1}{r_l} \, \partial G_C^N \left(\underline{u}_{ZN}^N \times \underline{v}^N \right)$$

$$\underline{g}_P^N = - H(R) \, \underline{u}_{ZN}^N + \partial g_{P_{Up}} \, \underline{u}_{ZN}^N + \partial g_{P_{North}} \, C_E^N \, \underline{u}_{North}^E$$

(Continued)

For $h \geq 0$:

$$H(R) = \frac{\mu}{R^2}$$

$$\partial g_{P_{Up}} \approx \frac{3}{2} J_2 \frac{\mu}{R^2} \left(\frac{R_0}{R}\right)^2 \left(3 \, u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R \, \omega_e^2$$

$$\partial g_{P_{North}} \approx u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left[6 \frac{\mu}{R_0^{\,3}} \left(2 J_2 - e\right) - \omega_e^2 \right] h$$

For $h < 0$:

$$H(R) = \frac{R}{R_S} \frac{\mu}{R_S^2} \qquad \text{(12.1.2-6) (Continued)}$$

$$\partial g_{P_{Up}} \approx \frac{R}{R_S} \left[\frac{3}{2} J_2 \frac{\mu}{R_S^2} \left(\frac{R_0}{R_S}\right)^2 \left(3 \, u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R_S \, \omega_e^2 \right]$$

$$\partial g_{P_{North}} \approx 0$$

$$\underline{u}_{ZN}^E = \begin{bmatrix} u_{ZN_{XE}} \\ u_{ZN_{YE}} \\ u_{ZN_{ZE}} \end{bmatrix} = C_N^E \, \underline{u}_{ZN}^N = \begin{bmatrix} D_{13} \\ D_{23} \\ D_{33} \end{bmatrix}$$

$$\underline{u}_{North}^E = \frac{1}{\sqrt{1 - u_{ZN_{YE}}^2}} \begin{bmatrix} - u_{ZN_{XE}} \, u_{ZN_{YE}} \\ 1 - u_{ZN_{YE}}^2 \\ - u_{ZN_{ZE}} \, u_{ZN_{YE}} \end{bmatrix}$$

12.1.3 NAVIGATION EQUATIONS FOR E FRAME ERROR ANALYSIS

For E Frame error analysis, the equivalent to Equation (12.1-1) is from generalized Equation (3.3.2-13):

$$\dot{C}_B^E = C_B^E \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IE}^E \times\right) C_B^E \qquad (12.1.3\text{-}1)$$

We also use Equation (12.1-14) with $\underline{a}_{SF}^E$ replaced by its equivalent transformation from the B Frame:

$$\dot{\underline{v}}^E = C_B^E \, \underline{a}_{SF}^B + \underline{g}_P^E - 2 \, \underline{\omega}_{IE}^E \times \underline{v}^E - e_{vc_1} \, \underline{u}_{ZN}^E \qquad (12.1.3\text{-}2)$$

The $\underline{g}_P^E$ term in (12.1.3-2) is obtained as the E Frame version of the equivalent expression in Equations (12.1.1-11):

$$\underline{g}_P^E = - H(R)\ \underline{u}_{ZN}^E + \partial g_{P_{Up}}\ \underline{u}_{ZN}^E + \partial g_{P_{North}}\ \underline{u}_{North}^E \tag{12.1.3-3}$$

With $\underline{u}_{ZN}^E$ from (12.1.1-14), Equation (12.1.3-3) becomes:

$$\begin{aligned} \underline{g}_P^E = & - \frac{H(R)}{R}\ \underline{R}^E + 2\ e\ H(R)\ \ u_{ZN_{YE}} \frac{R_0}{R} \left(u_{ZN_{YE}} \frac{1}{R}\ \underline{R}^E - \underline{u}_{YE}^E \right) \\ & + \partial g_{P_{Up}}\ \underline{u}_{ZN}^E + \partial g_{P_{North}}\ \underline{u}_{North}^E \end{aligned} \tag{12.1.3-4}$$

Components of $\underline{u}_{ZN}^E$ appearing in (12.1.3-4) and other equations are provided by a first order (in e) Picard type approximation to (12.1.1-14) in which the $\underline{u}_{ZN}^E$ components in the term multiplying e are approximated by the (12.1.1-14) zero order solution: $\underline{u}_{ZN}^E \approx \underline{R}^E /\ R$. With this substitution, (12.1.1-14) becomes:

$$\underline{u}_{ZN}^E = \begin{bmatrix} u_{ZN_{XE}} \\ u_{ZN_{YE}} \\ u_{ZN_{ZE}} \end{bmatrix} \approx \frac{1}{R}\ \underline{R}^E - 2\ e\ \frac{R_{YE}\ R_0}{R^2} \left(\frac{R_{YE}}{R^2}\ \underline{R}^E - \underline{u}_{YE}^E \right) \tag{12.1.3-5}$$

where

R_{YE} = E Frame Y axis component of $\underline{R}^E$.

Finally, the position parameter for E Frame navigation equation analysis will be the position vector $\underline{R}^E$ using Equation (12.1-13), hence, the position vector magnitude R is computed as:

$$R = \sqrt{\underline{R}^E \cdot \underline{R}^E} \tag{12.1.3-6}$$

Substituting Equations (12.1.3-1) - (12.1.3-2) and (12.1.3-4) - (12.1.3-6) for the equivalent terms in Equations (12.1-1) - (12.1-13) and (12.1.1-13), with retention of applicable terms from Equations (12.1-1) - (12.1-13), (12.1-33), (12.1-36), and (12.1.1-13) then provides the desired navigation equation set for subsequent E Frame error analysis:

$$\dot{C}_B^E = C_B^E \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IE}^E \times\right) C_B^E$$

$$\underline{\omega}_{IE}^E = \left[\ 0\ \ \omega_e\ \ 0\ \right]^T$$

$$\dot{\underline{v}}^E = C_B^E\, \underline{a}_{SF}^B + \underline{g}_P^E - 2\, \underline{\omega}_{IE}^E \times \underline{v}^E - e_{vc_1}\, \underline{u}_{ZN}^E$$

$$\dot{\underline{R}}^E = \underline{v}^E - e_{vc_2}\, \underline{u}_{ZN}^E$$

$$R = \sqrt{\underline{R}^E \cdot \underline{R}^E}$$

$$R_S \approx \left(1 - u_{ZN_{YE}}^2\, e\right) R_0$$

$$h = R - R_S$$

$$\partial h = h - h_{Prsr}$$

$$e_{vc_1} = e_{vc_3} + C_2\, \partial h \qquad e_{vc_2} = C_3\, \partial h \qquad \dot{e}_{vc_3} = C_1\, \partial h \tag{12.1.3-7}$$

$$\underline{g}_P^E = -\, H(R)\, \frac{1}{R}\, \underline{R}^E + 2\, e\, H(R)\ u_{ZN_{YE}}\, \frac{R_0}{R} \left(u_{ZN_{YE}}\, \frac{1}{R}\, \underline{R}^E - \underline{u}_{YE}^E\right)$$
$$+\ \partial g_{P_{Up}}\, \underline{u}_{ZN}^E + \partial g_{P_{North}}\, \underline{u}_{North}^E$$

For $h \geq 0$:

$$H(R) = \frac{\mu}{R^2}$$

$$\partial g_{P_{Up}} \approx \frac{3}{2}\, J_2\, \frac{\mu}{R^2} \left(\frac{R_0}{R}\right)^2 \left(3\, u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R\, \omega_e^2$$

$$\partial g_{P_{North}} \approx u_{ZN_{YE}} \sqrt{1 - u_{ZN_{YE}}^2} \left[6\, \frac{\mu}{{R_0}^3} \left(2\, J_2 - e\right) - \omega_e^2\right] h$$

(Continued)

For $h < 0$:

$$H(R) = \frac{R}{R_S}\frac{\mu}{R_S^2}$$

$$\partial g_{P_{Up}} \approx \frac{R}{R_S}\left[\frac{3}{2} J_2 \frac{\mu}{R_S^2}\left(\frac{R_0}{R_S}\right)^2 \left(3\, u_{ZN_{YE}}^2 - 1\right) + \left(1 - u_{ZN_{YE}}^2\right) R_S\, \omega_e^2\right]$$

$$\partial g_{P_{North}} \approx 0$$

$$\underline{u}_{ZN}^E = \begin{bmatrix} u_{ZN_{XE}} \\ u_{ZN_{YE}} \\ u_{ZN_{ZE}} \end{bmatrix} \approx \frac{1}{R}\, \underline{R}^E - 2\, e\, \frac{R_{YE}\, R_0}{R^2}\left(\frac{R_{YE}}{R^2}\, \underline{R}^E - \underline{u}_{YE}^E\right) \tag{12.1.3-7 (Continued)}$$

$$\underline{u}_{North}^E = \frac{1}{\sqrt{1 - u_{ZN_{YE}}^2}}\begin{bmatrix} - u_{ZN_{XE}}\, u_{ZN_{YE}} \\ 1 - u_{ZN_{YE}}^2 \\ - u_{ZN_{ZE}}\, u_{ZN_{YE}} \end{bmatrix}$$

$$\underline{u}_{YE}^E = \left[\ 0 \ \ 1 \ \ 0\ \right]^T$$

12.1.4 NAVIGATION EQUATIONS FOR I FRAME ERROR ANALYSIS

For error analysis purposes and for some specialized applications it is useful to have a set of navigation equations based on motion relative to non-rotating inertial space (i.e., the I Frame). In this section we develop such a set beginning with generalized equation (3.3.2-6) applied to B Frame attitude determination relative to the I Frame:

$$\dot{C}_B^I = C_B^I\left(\underline{\omega}_{IB}^B \times\right) \tag{12.1.4-1}$$

where

C_B^I = Direction cosine matrix that transforms vectors from the B Frame to the I Frame.

$\underline{\omega}_{IB}^B$ = Angular rate of the B Frame relative to the I Frame as projected on B Frame axes (i.e., the angular rate sensed by strapdown angular rate sensors).

We define a velocity vector for I Frame analysis as the time rate of change of position measured in the I Frame:

$$\underline{\upsilon}^I \equiv \dot{\underline{R}}^I \tag{12.1.4-2}$$

where

$\underline{\upsilon}^I$ = Velocity relative to non-rotating inertial space (i.e., the I Frame) as projected (superscript designator) on I Frame axes.

$\underline{R}^I$ = Position vector from earth's center to the navigation system projected on I Frame axes.

In contrast, the velocity vector relative to the earth E Frame was defined by Equation (4.3-1) as:

$$\underline{v}^E \equiv \dot{\underline{R}}^E \tag{12.1.4-3}$$

where

$\underline{v}^E$ = Velocity relative to the earth (i.e., the E Frame) as projected on E Frame axes.

But we also know from generalized vector rate transformation Equation (3.4-6) that:

$$\dot{\underline{R}}^I = C_E^I\, \dot{\underline{R}}^E + \underline{\omega}_{IE}^I \times \underline{R}^I \tag{12.1.4-4}$$

Substitution of (12.1.4-2) and (12.1.4-3) into (12.1.4-4) provides the relationship between the $\underline{\upsilon}$ and $\underline{v}$ velocity parameters:

$$\underline{\upsilon}^I = \underline{v}^I + \underline{\omega}_{IE}^I \times \underline{R}^I \tag{12.1.4-5}$$

The time rate of change of $\underline{\upsilon}^I$ is determined from the time derivative of Equation (12.1.4-2):

$$\dot{\underline{\upsilon}}^I = \ddot{\underline{R}}^I \tag{12.1.4-6}$$

From Equation (4.3-11), the $\ddot{\underline{R}}^I$ term in (12.1.4-6) is given by:

$$\ddot{\underline{R}}^I = \underline{g}^I + \underline{a}_{SF}^I \tag{12.1.4-7}$$

where

$\underline{g}^I$ = Earth's mass attraction gravity vector projected on I Frame axes.

$\underline{a}_{SF}^I$ = Specific force acceleration projected on I Frame axes.

Equation (4.3-15) shows that:

$$\underline{g}^I = \underline{g}_P^I + \underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \underline{R}^I\right) \tag{12.1.4-8}$$

where

$\underline{g}_P^I$ = Plumb-bob gravity projected on I Frame axes.

The I Frame has been defined to have its Y axis along the earth's polar (i.e., rotation) axis, hence, we can write for $\underline{\omega}_{IE}^I$ in (12.1.4-8):

$$\underline{\omega}_{IE}^I = \left[\; 0 \;\; \omega_e \;\; 0 \;\right]^T \tag{12.1.4-9}$$

where

ω_e = Earth's inertial rotation rate magnitude.

With (12.1.4-7), (12.1.4-8) and expansion of $\underline{a}_{SF}^I$, we obtain the desired expression for $\dot{\underline{\upsilon}}^I$. Including vertical channel control as in (12.1-14), but in the I Frame, the result is:

$$\dot{\underline{\upsilon}}^I = C_B^I \; \underline{a}_{SF}^B + \underline{g}_P^I + \underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \underline{R}^I\right) - e_{vc_1} \, \underline{u}_{ZN}^I \tag{12.1.4-10}$$

where

$\underline{a}_{SF}^B$ = Specific force acceleration projected on B Frame axes (i.e., the acceleration measured by strapdown accelerometers).

The $u_{ZN_{YE}}$ term appearing extensively in Equations (12.1.3-7) is defined as the E Frame Y axis component of $\underline{u}_{ZN}$, the unit vector along the N Frame Z axis (along the upward local geodetic vertical). From their definitions, the E and I Frames have common Y axes (along the earth's polar axis), thus:

$$\underline{u}_{YE} = \underline{u}_{YI} \qquad\qquad u_{ZN_{YE}} = u_{ZN_{YI}} \tag{12.1.4-11}$$

where

$\underline{u}_{YE}, \underline{u}_{YI}$ = Unit vectors along the E and I Frame Y axes.

$u_{ZN_{YI}}$ = I Frame Y axis component of the $\underline{u}_{ZN}$ unit vector along the N Frame Z axis.

Applying (12.1.4-11), the $\underline{g}_P^I$ term in (12.1.4-10) is provided from Equations (12.1.3-7) transformed to the I Frame:

$$\underline{g}_P^I = - H(R) \, \frac{1}{R} \, \underline{R}^I + 2 \, e \, H(R) \, u_{ZN\,YI} \, \frac{R_0}{R} \left(u_{ZN\,YI} \, \frac{1}{R} \, \underline{R}^I - \underline{u}_{YI}^I \right)$$
$$+ \, \partial g_{P_{Up}} \, \underline{u}_{ZN}^I + \partial g_{P_{North}} \, \underline{u}_{North}^I$$

For $h \geq 0$:

$$H(R) = \frac{\mu}{R^2}$$

$$\partial g_{P_{Up}} \approx \frac{3}{2} \, J_2 \, \frac{\mu}{R^2} \left(\frac{R_0}{R} \right)^2 \left(3 \, u_{ZN\,YI}^2 - 1 \right) + \left(1 - u_{ZN\,YI}^2 \right) R \, \omega_e^2 \tag{12.1.4-12}$$

$$\partial g_{P_{North}} \approx u_{ZN\,YI} \sqrt{1 - u_{ZN\,YI}^2} \left[6 \, \frac{\mu}{R_0^3} \left(2 \, J_2 - e \right) - \omega_e^2 \right] h$$

For $h < 0$:

$$H(R) = \frac{R}{R_S} \, \frac{\mu}{R_S^2}$$

$$\partial g_{P_{Up}} \approx \frac{R}{R_S} \left[\frac{3}{2} \, J_2 \, \frac{\mu}{R_S^2} \left(\frac{R_0}{R_S} \right)^2 \left(3 \, u_{ZN\,YI}^2 - 1 \right) + \left(1 - u_{ZN\,YI}^2 \right) R_S \, \omega_e^2 \right]$$

$$\partial g_{P_{North}} \approx 0$$

From its definition we also write:

$$\underline{u}_{YI}^I = \left[\; 0 \;\; 1 \;\; 0 \; \right]^T \tag{12.1.4-13}$$

Components of $\underline{u}_{ZN}^I$ appearing in (12.1.4-12) and other equations are determined by transforming Equation (12.1.1-14) to the I Frame, applying (12.1.4-11), and approximating the result as a first order Picard expansion (as in (12.1.3-5)) in which the $\underline{u}_{ZN}^I$ components in the term multiplying e are approximated by the zero order solution: $\underline{u}_{ZN}^I \approx \underline{R}^I \, / \, R$. Thus:

$$\underline{u}_{ZN}^I = \begin{bmatrix} u_{ZN\,XI} \\ u_{ZN\,YI} \\ u_{ZN\,ZI} \end{bmatrix} \approx \frac{1}{R} \, \underline{R}^I - 2 \, e \, \frac{R_{YI} \, R_0}{R^2} \left(\frac{R_{YI}}{R^2} \, \underline{R}^I - \underline{u}_{YI}^I \right) \tag{12.1.4-14}$$

where

$u_{ZN\,XI}, u_{ZN\,YI}, u_{ZN\,ZI}$ = I Frame X, Y, Z components of $\underline{u}_{ZN}^I$.

R_{YI} = I Frame Y axis component of $\underline{R}^I$.

The $\underline{u}^{I}_{North}$ term in (12.1.4-12) is derived in the I Frame using the I Frame version of $\underline{u}^{E}_{North}$ in Equation (12.1-36):

$$\underline{u}^{I}_{North} = \frac{1}{\sqrt{1 - u^{2}_{ZN_{YI}}}} \begin{bmatrix} - u_{ZN_{XI}}\, u_{ZN_{YI}} \\ 1 - u^{2}_{ZN_{YI}} \\ - u_{ZN_{ZI}}\, u_{ZN_{YI}} \end{bmatrix} \qquad (12.1.4\text{-}15)$$

Finally, the I Frame position rate equation is provided simply by Equation (12.1.4-2) including vertical channel control as in (12.1-13), but in the I Frame:

$$\dot{\underline{R}}^{I} = \underline{\upsilon}^{I} - e_{vc_2}\, \underline{u}^{I}_{ZN} \qquad (12.1.4\text{-}16)$$

with the magnitude of $\underline{R}^{I}$ identified as R:

$$R = \sqrt{\underline{R}^{I} \cdot \underline{R}^{I}} \qquad (12.1.4\text{-}17)$$

In summary, the equations for subsequent I Frame navigation error analysis are given by Equations (12.1.4-1), (12.1.4-9) - (12.1.4-10), and (12.1.4-12) - (12.1.4-17) with R_S and h calculated as in (12.1.3-7) using (12.1.4-11). The equations are repeated below for easy reference:

$$\begin{aligned}
\dot{C}^{I}_{B} &= C^{I}_{B}\left(\underline{\omega}^{B}_{IB} \times\right) \\
\dot{\underline{\upsilon}}^{I} &= C^{I}_{B}\, \underline{a}^{B}_{SF} + \underline{g}^{I}_{P} + \underline{\omega}^{I}_{IE} \times \left(\underline{\omega}^{I}_{IE} \times \underline{R}^{I}\right) - e_{vc_1}\, \underline{u}^{I}_{ZN} \\
\dot{\underline{R}}^{I} &= \underline{\upsilon}^{I} - e_{vc_2}\, \underline{u}^{I}_{ZN} \\
\underline{\omega}^{I}_{IE} &= \left[\, 0 \;\; \omega_e \;\; 0 \,\right]^{T} \\
R &= \sqrt{\underline{R}^{I} \cdot \underline{R}^{I}} \\
R_S &\approx \left(1 - u^{2}_{ZN_{YI}}\, e\right) R_0 \\
h &= R - R_S
\end{aligned} \qquad (12.1.4\text{-}18)$$

(Continued)

$$\partial h \;=\; h - h_{Prsr}$$

$$e_{vc_1} \;=\; e_{vc_3} + C_2\,\partial h \qquad e_{vc_2} \;=\; C_3\,\partial h \qquad \dot{e}_{vc_3} \;=\; C_1\,\partial h$$

$$\underline{g}_P^I \;=\; -\,H(R)\,\frac{1}{R}\,\underline{R}^I + 2\,e\,H(R)\;u_{ZN_{YI}}\,\frac{R_0}{R}\left(u_{ZN_{YI}}\,\frac{1}{R}\,\underline{R}^I - \underline{u}_{YI}^I\right)$$
$$+\,\partial g_{P_{Up}}\,\underline{u}_{ZN}^I + \partial g_{P_{North}}\,\underline{u}_{North}^I$$

For $h \;\geq\; 0$:

$$H(R) \;=\; \frac{\mu}{R^2}$$

$$\partial g_{P_{Up}} \;\approx\; \frac{3}{2}\,J_2\,\frac{\mu}{R^2}\left(\frac{R_0}{R}\right)^2\left(3\,u_{ZN_{YI}}^2 - 1\right) + \left(1 - u_{ZN_{YI}}^2\right)R\,\omega_e^2$$

$$\partial g_{P_{North}} \;\approx\; u_{ZN_{YI}}\sqrt{1 - u_{ZN_{YI}}^2}\left[6\,\frac{\mu}{R_0{}^3}\left(2\,J_2 - e\right) - \omega_e^2\right]h$$

For $h \;<\; 0$:

$$H(R) \;=\; \frac{R}{R_S}\,\frac{\mu}{R_S^2} \tag{12.1.4-18 (Continued)}$$

$$\partial g_{P_{Up}} \;\approx\; \frac{R}{R_S}\left[\frac{3}{2}\,J_2\,\frac{\mu}{R_S^2}\left(\frac{R_0}{R_S}\right)^2\left(3\,u_{ZN_{YI}}^2 - 1\right) + \left(1 - u_{ZN_{YI}}^2\right)R_S\,\omega_e^2\right]$$

$$\partial g_{P_{North}} \;\approx\; 0$$

$$\underline{u}_{ZN}^I \;=\; \begin{bmatrix} u_{ZN_{XI}} \\ u_{ZN_{YI}} \\ u_{ZN_{ZI}} \end{bmatrix} \;\approx\; \frac{1}{R}\,\underline{R}^I - 2\,e\,\frac{R_{YI}\;R_0}{R^2}\left(\frac{R_{YI}}{R^2}\,\underline{R}^I - \underline{u}_{YI}^I\right)$$

$$\underline{u}_{North}^I \;=\; \frac{1}{\sqrt{1 - u_{ZN_{YI}}^2}}\begin{bmatrix} -\,u_{ZN_{XI}}\,u_{ZN_{YI}} \\ 1 - u_{ZN_{YI}}^2 \\ -\,u_{ZN_{ZI}}\,u_{ZN_{YI}} \end{bmatrix}$$

$$\underline{u}_{YI}^I \;=\; \left[\;0 \;\; 1 \;\; 0\;\right]^T$$

12.2 NAVIGATION ERROR PARAMETERS

In this section we provide analytical definitions for several commonly used attitude, velocity and position navigation error parameters, their relationship to the inertial navigation parameters calculated by the inertial integration process, and analytical equivalencies between different error parameters. We will also discuss analytical error forms of the gravity and transport rate vectors for subsequent application in the derivation of navigation error parameter time rate differential equations. The section concludes with a general discussion of the process of navigation parameter selection for a particular application.

12.2.1 ANGULAR ERROR PARAMETERS

In typical strapdown inertial navigation systems, angular data (e.g., in the form of direction cosine matrices) are calculated that relate the B Frame to the L Frame and the N Frame to the E Frame. The direction cosine matrix relating the B and E Frames can also be computed as:

$$C_B^N = C_L^N \, C_B^L \tag{12.2.1-1}$$

$$C_B^E = C_N^E \, C_B^N \tag{12.2.1-2}$$

where

C_B^L = Direction cosine matrix that transforms vectors from the B Frame to the L Frame.

C_L^N = Direction cosine matrix that transforms vectors from the L Frame to the N Frame (a constant matrix as defined in Equation (4.1.1-2).

C_N^E = Direction cosine matrix that transforms vectors from the N Frame to the E Frame.

C_B^E = Direction cosine matrix that transforms vectors from the B Frame to the E Frame.

The error in the C_B^E matrix can be defined as a rotation error vector by applying generalized error Equations (3.5.2-27):

$$\delta C_B^E = - \left(\underline{\psi}^E \times \right) C_B^E \tag{12.2.1-3}$$

$$\left(\underline{\psi}^E \times \right) \equiv I - \hat{C}_B^E \left(C_B^E \right)^T \tag{12.2.1-4}$$

where

$\widehat{\ }$ = Designation for parameter calculated in the strapdown system computer that may contain errors. The parameter without the $\widehat{\ }$ is defined to be the idealized error free value.

$\hat{C}_B^E$ = Strapdown system computer calculated C_B^E matrix (i.e., containing errors).

$\underline{\psi}^E$ = Rotation angle error vector associated with the $\hat{C}_B^E$ matrix considering the E Frame to be misaligned, as projected on Frame E axes.

δC_B^E = Error in $\hat{C}_B^E$ caused by misalignment.

Equations (12.2.1-3) - (12.2.1-4) are based on the assumption that the errors in $\hat{C}_B^E$ are not caused by orthogonality and/or normality error. This same assumption will be used throughout this section for all attitude error parameters analyzed. Careful software design and validation will assure this assumption to be valid. Section 3.5.1 discusses the general case in which the direction cosine matrix may contain orthogonality, normality and misalignment errors.

Substituting (12.2.1-2) into (12.2.1-3) yields:

$$\delta C_B^E = - C_N^E \, C_E^N \left(\underline{\psi}^E \times\right) \left(C_E^N\right)^T C_B^N \tag{12.2.1-5}$$

or after identifying the middle term as a similarity transformation of $\left(\underline{\psi}^E \times\right)$ (as in generalized Equation (3.1.1-39)):

$$\delta C_B^E = - C_N^E \left(\underline{\psi}^N \times\right) C_B^N \tag{12.2.1-6}$$

with

$$\underline{\psi}^N = C_E^N \, \underline{\psi}^E \tag{12.2.1-7}$$

The system computer version of Equation (12.2.1-2) is:

$$\hat{C}_B^E = \hat{C}_N^E \, \hat{C}_B^N \tag{12.2.1-8}$$

where

$\hat{C}_N^E, \hat{C}_B^N$ = Computer calculated values for C_N^E, C_B^N (containing errors).

The error characteristics of $\hat{C}_N^E$ and $\hat{C}_B^N$ can also be defined by rotation error vectors through application of generalized Equations (3.5.2-27) and (3.5.2-28):

$$\delta C_B^N = - \left(\underline{\gamma}^N \times \right) C_B^N \tag{12.2.1-9}$$

$$\left(\underline{\gamma}^N \times \right) = I - \hat{C}_B^N \left(C_B^N \right)^T \tag{12.2.1-10}$$

$$\delta C_N^E = C_N^E \left(\underline{\varepsilon}^N \times \right) \tag{12.2.1-11}$$

$$\left(\underline{\varepsilon}^N \times \right) = \left(C_N^E \right)^T \hat{C}_N^E - I \tag{12.2.1-12}$$

where

$\underline{\gamma}^N$ = Rotation angle error vector associated with the $\hat{C}_B^N$ matrix considering the N Frame to be misaligned, as projected on Frame N axes.

$\underline{\varepsilon}^N$ = Rotation angle error vector associated with the $\hat{C}_N^E$ matrix considering the N Frame to be misaligned, as projected on Frame N axes.

Equation (12.2.1-8) has the same form as general Equation (3.5.2-40) repeated below:

$$\hat{C}_D^A = \hat{C}_B^A \hat{C}_D^B \tag{12.2.1-13}$$

for which general Equation (3.5.2-45) applies as follows:

$$\underline{\alpha}_{D\,\mathrm{to}\,A}^A = \underline{\alpha}_{D\,\mathrm{to}\,B}^A + \underline{\alpha}_{B\,\mathrm{to}\,A}^A \tag{12.2.1-14}$$

or with general Equations (3.5.2-31):

$$\underline{\alpha}_{D\,\mathrm{to}\,A}^A = \underline{\alpha}_{D\,\mathrm{to}\,B}^A - \underline{\beta}_{B\,\mathrm{to}\,A}^A \tag{12.2.1-15}$$

where

D, B, A = Arbitrary coordinate frames.

$\hat{C}_D^A$, $\hat{C}_B^A$, $\hat{C}_D^B$ = Direction cosine matrices calculated in the strapdown system computer that transform vectors from the D to A, B to A and D to B Frame.

$\underline{\alpha}_{D\,\mathrm{to}\,A}^A$ = Rotation angle error vector associated with the $\hat{C}_D^A$ matrix considering the A Frame to be misaligned, as projected onto A Frame axes.

$\underline{\beta}_{B\,\mathrm{to}\,A}^A$ = Rotation angle error vector associated with the $\hat{C}_B^A$ matrix considering the B Frame to be misaligned, as projected onto A Frame axes.

$\underline{\alpha}^{A}_{D\,to\,B}$ = Rotation angle error vector associated with the $\hat{C}^{B}_{D}$ matrix considering the B Frame to be misaligned, as projected onto A Frame axes.

Applying general Equations (12.2.1-13) and (12.2.1-15) to Equation (12.2.1-8) and the definitions for the associated rotation error vectors $\underline{\psi}$, $\underline{\varepsilon}$ and $\underline{\gamma}$ finds that:

$$\underline{\psi}^{E} = \underline{\gamma}^{E} - \underline{\varepsilon}^{E} \tag{12.2.1-16}$$

or in the N Frame:

$$\underline{\psi}^{N} = \underline{\gamma}^{N} - \underline{\varepsilon}^{N} \tag{12.2.1-17}$$

Equation (12.2.1-17) shows the equivalency between the $\underline{\psi}$, $\underline{\varepsilon}$ and $\underline{\gamma}$ error angle vectors.

As an aside, it is noted that Equation (12.2.1-17) is equivalent to Equation (7.9) of Reference 15 (a chapter in one of the original inertial navigation textbooks). The $\delta\underline{\theta}$ "computer angle error", $\underline{\phi}$ "platform angle error" and $\underline{\psi}$ "angular error between computer and platform axes" of Reference 15 have identical meanings as the $\underline{\varepsilon}$, $\underline{\gamma}$ and $\underline{\psi}$ errors of this section (defined analytically by Equations (12.2.1-3) - (12.2.1-4) and (12.2.1-9) - (12.2.1-12)).

For error analysis purposes, we will at times make use of error parameters defined in the inertial non-rotating I Frame. We show here, that the $\underline{\psi}$ attitude error parameter defined in the E Frame by Equation (12.2.1-4) is identical if defined in the I Frame. To do this, we temporarily identify the attitude error defined in the I Frame as:

$$\left(\underline{\varphi}^{I}\times\right) \equiv I - \hat{C}^{I}_{B}\left(C^{I}_{B}\right)^{T} \tag{12.2.1-18}$$

where

C^{I}_{B} = Direction cosine matrix that transforms vectors from the B Frame to the I Frame.

$\hat{C}^{I}_{B}$ = Strapdown system computer calculated C^{I}_{B} matrix (i.e., containing errors).

$\underline{\varphi}^{I}$ = Rotation angle error vector associated with the $\hat{C}^{I}_{B}$ matrix considering the I Frame in $\hat{C}^{I}_{B}$ to be misaligned (relative to the B Frame), as projected on I Frame axes.

We also write:

$$C^{I}_{B} = C^{I}_{E}\,C^{E}_{B} \qquad \hat{C}^{I}_{B} = \hat{C}^{I}_{E}\,\hat{C}^{E}_{B} \tag{12.2.1-19}$$

where

C_E^I = Direction cosine matrix that transforms vectors from the E Frame to the I Frame.

The C_E^I matrix is created by the integrated effect of earth's rotation rate which is constant in the E and I Frames (along the E and I Frame Y axes that have been defined to be coincident). Because earth rate is a known constant, there is no error in the system software value for C_E^I, hence:

$$\hat{C}_E^I = C_E^I \tag{12.2.1-20}$$

Substituting (12.2.1-20) and (12.2.1-19) into (12.2.1-18) gives:

$$\left(\underline{\varphi}^I \times\right) = I - C_E^I \, \hat{C}_B^E \left(C_B^E\right)^T \left(C_E^I\right)^T \tag{12.2.1-21}$$

Multiplying (12.2.1-21) on the left by $\left(C_E^I\right)^T$ and on the right by C_E^I, and using Equation (3.2.1-3) then obtains:

$$\left(C_E^I\right)^T \left(\underline{\varphi}^I \times\right) C_E^I = C_I^E \left(\underline{\varphi}^I \times\right) \left(C_I^E\right)^T = I - \hat{C}_B^E \left(C_B^E\right)^T \tag{12.2.1-22}$$

or with (3.1.1-39):

$$\left(\underline{\varphi}^E \times\right) = I - \hat{C}_B^E \left(C_B^E\right)^T \tag{12.2.1-23}$$

Comparing (12.2.1-23) with (12.2.1-4) we see then as stipulated that:

$$\underline{\varphi}^E = \underline{\psi}^E \tag{12.2.1-24}$$

We conclude this section with a discussion of the errors in the Euler angles associated with the $\hat{C}_B^L$ matrix and their relationship with the other angle error parameters. As in Section 3.2.3 and 3.2.3.1, we first define the Euler angle sequence for C_B^L in terms of three successive rotations about intermediate frames:

$$C_B^L = C_{L_1}^L \, C_{L_2}^{L_1} \, C_B^{L_2} \tag{12.2.1-25}$$

where

L_1 Frame = Frame L after rotating it about axis Z through the heading Euler angle.

L_2 Frame = Frame L_1 after rotating it about axis Y through the pitch Euler angle.

B Frame = Strapdown sensor axis B Frame (as defined in Section 2.2) obtained by rotating Frame L_2 about the X-axis through the roll Euler angle.

Applying general Equations (3.5.2-27), (3.5.3-1), (3.5.3-25) - (3.5.3-26), (3.5.3-28) and (3.5.3-35) to Equation (12.2.1-25) shows that:

$$\left(\underline{\alpha}_{B\,to\,L}^{L} \times\right) = I - \hat{C}_{B}^{L}\left(C_{B}^{L}\right)^{T} \qquad (12.2.1\text{-}26)$$

$$\underline{\alpha}_{B\,to\,L}^{L} = -\,\delta\psi_P\,\underline{u}_{\zeta L}^{L} - \delta\theta\,\underline{u}_{\zeta L_1}^{L} - \delta\phi\,\underline{u}_{\zeta L_2}^{L} \qquad (12.2.1\text{-}27)$$

$$\delta\psi_P \equiv \hat{\psi}_P - \psi_P \qquad \delta\theta \equiv \hat{\theta} - \theta \qquad \delta\phi \equiv \hat{\phi} - \phi$$

$$\underline{u}_{\zeta L_1}^{L} = C_{L_1}^{L}\,\underline{u}_{\zeta L_1}^{L_1} \qquad \underline{u}_{\zeta L_2}^{L} = C_{L_1}^{L}\,C_{L_2}^{L_1}\,\underline{u}_{\zeta L_2}^{L_2}$$

$$\underline{u}_{\zeta L}^{L} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \underline{u}_{\zeta L_1}^{L_1} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \underline{u}_{\zeta L_2}^{L_2} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad (12.2.1\text{-}28)$$

$$C_{L_1}^{L} = \begin{bmatrix} \cos\psi_P & -\sin\psi_P & 0 \\ \sin\psi_P & \cos\psi_P & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad C_{L_2}^{L_1} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$C_{B}^{L_2} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$

where

$\underline{\alpha}_{B\,to\,L}^{L}$ = Rotation angle error vector associated with $\hat{C}_{B}^{L}$ considering the L Frame to be misaligned, as projected on L Frame axes.

$\underline{u}_{\zeta L_2}^{L_2}$, $\underline{u}_{\zeta L_1}^{L_1}$, $\underline{u}_{\zeta L}^{L}$ = Unit vectors along the L_2 Frame X-axis, L_1 Frame Y-axis, L Frame Z-axis.

ϕ, θ, ψ_P = Roll, pitch, "platform" heading, Euler angles associated with C_B^L around $\underline{u}_{\zeta L_2}^{L_2}$, $\underline{u}_{\zeta L_1}^{L_1}$, $\underline{u}_{\zeta L}^{L}$. The term "platform" heading is commonly used to describe this heading Euler angle which is referenced to local level L Frame axes (as contrasted with "true" heading which is the heading relative to true north). Platform heading is the heading of the B Frame relative to an L Frame identified as "platform coordinates" (analogous to a gimbaled stable platform aligned with the L frame - See Chapter 1).

$\delta\phi, \delta\theta, \delta\psi_P$ = Errors in the system computer calculated values for ϕ, θ, ψ_P.

Equation (12.2.1-26) can be converted to the equivalent N Frame form by pre-multiplying by C_L^N, post-multiplying by $\left(C_L^N\right)^T$, and applying Equation (3.2.1-3) and general similarity transformation Equation (3.1.1-39) for cross-product operators:

$$
\begin{aligned}
C_L^N \left(\underline{\alpha}_{B to L}^L \times\right) \left(C_L^N\right)^T &= \left(\underline{\alpha}_{B to L}^N \times\right) \\
&= C_L^N \left(C_L^N\right)^T - C_L^N \hat{C}_B^L \left(C_B^L\right)^T \left(C_L^N\right)^T = I - C_L^N \hat{C}_B^L \left(C_L^N \, C_B^L\right)^T
\end{aligned}
\tag{12.2.1-29}
$$

But because C_L^N is a constant, $C_L^N = \hat{C}_L^N$, and:

$$
C_L^N \, \hat{C}_B^L = \hat{C}_B^N \tag{12.2.1-30}
$$

Hence, Equation (12.2.1-29) becomes:

$$
\left(\underline{\alpha}_{B to L}^N \times\right) = I - \hat{C}_B^N \left(C_B^N\right)^T \tag{12.2.1-31}
$$

with

$$
\underline{\alpha}_{B to L}^N = C_L^N \, \underline{\alpha}_{B to L}^L \tag{12.2.1-32}
$$

If we now compare Equations (12.2.1-31) and (12.2.1-10) it should be clear as might have been expected that:

$$
\underline{\alpha}_{B to L}^N = \underline{\gamma}^N \tag{12.2.1-33}
$$

Substituting (12.2.1-33) and (12.2.1-27) into (12.2.1-32) then obtains the desired expression for $\underline{\gamma}^N$ as a function of the Euler angle errors:

$$
\underline{\gamma}^N = - C_L^N \left(\delta\psi_P \, \underline{u}_{\zeta L}^L + \delta\theta \, \underline{u}_{\zeta L_1}^L + \delta\phi \, \underline{u}_{\zeta L_2}^L\right) \tag{12.2.1-34}
$$

with $\underline{u}^L_{\zeta L}$, $\underline{u}^L_{\zeta L_1}$, $\underline{u}^L_{\zeta L_2}$ as defined in Equations (12.2.1-28) and C^N_L (the transpose of C^L_N) defined from Equation (4.1.1-2) as:

$$C^N_L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{12.2.1-35}$$

Equation (12.2.1-34) with (12.2.1-28) and (12.2.1-35) lends itself nicely for component evaluation by the "Method of Least Work" technique described in Section 3.2.3.3 as provided in Figure 12.2.1-1:

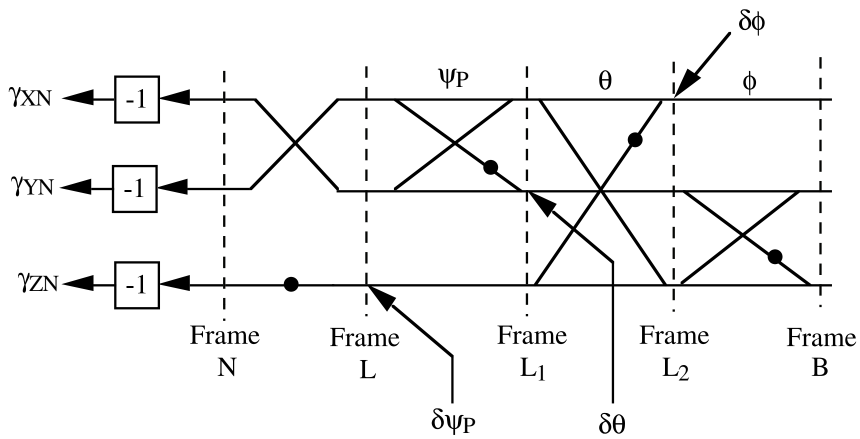


Figure 12.2.1-1 Rotation Angle Error As Function Of Euler Angle Errors

where

$\gamma_{XN}, \gamma_{YN}, \gamma_{ZN}$ = N Frame X, Y, Z components of $\underline{\gamma}^N$.

From Figure 12.2.1-1 we can immediately write:

$$\begin{aligned} \gamma_{XN} &= - \delta\phi \cos\theta \sin\psi_P - \delta\theta \cos\psi_P \\ \gamma_{YN} &= - \delta\phi \cos\theta \cos\psi_P + \delta\theta \sin\psi_P \\ \gamma_{ZN} &= - \delta\phi \sin\theta + \delta\psi_P \end{aligned} \tag{12.2.1-36}$$

The inverse of Equations (12.2.1-36) is obtained from Figure 12.2.1-1 by first writing the node equations at Frame L_1 for the X, Y components by reversing the arrows on the left of Frame L_1 and equating inputs from the left to inputs from the right:

$$
\begin{aligned}
& -\gamma_{XN} \sin \psi_P - \gamma_{YN} \cos \psi_P = \delta\phi \cos \theta \\
& -\gamma_{XN} \cos \psi_P + \gamma_{YN} \sin \psi_P = \delta\theta
\end{aligned} \tag{12.2.1-37}
$$

Equations (12.2.1-37) are easily inverted to find $\delta\phi$, $\delta\theta$ as a function of the $\underline{\gamma}^N$ components. The $\delta\psi_P$ Euler angle error is then obtained by substitution of the resulting $\delta\phi$ solution into the Equations (12.2.1-36) γ_{ZN} expression and rearranging. The result is:

$$
\begin{aligned}
\delta\phi &= -\sec \theta \left(\gamma_{XN} \sin \psi_P + \gamma_{YN} \cos \psi_P \right) \\
\delta\theta &= \gamma_{YN} \sin \psi_P - \gamma_{XN} \cos \psi_P \\
\delta\psi_P &= \gamma_{ZN} - \tan \theta \left(\gamma_{XN} \sin \psi_P + \gamma_{YN} \cos \psi_P \right)
\end{aligned} \tag{12.2.1-38}
$$

True heading is defined as the angle around the local vertical (measured in the positive sense for positive rotations around a downward vertical) from a horizontal true north pointing line (i.e., toward the earth positive polar axis) to the vertical projection of the L Frame X-axis on the local horizontal plane. For the special case when the X-axis of the L Frame points north, the "platform heading" angle ψ_P would equal the true heading. In the general case, the angle from true north to the L Frame X-axis (measured positive as a positive rotation about an upward vertical) is denoted as the "wander angle", and true heading is related to platform heading by the Equation (4.1.2-2) expression:

$$
\psi_T = \psi_P - \alpha \tag{12.2.1-39}
$$

where

α = Wander angle equal to the angle about the local vertical (measured positive for positive rotations about an upward vertical) from horizontal true north to the L Frame X-axis.

ψ_T = True heading.

The error form of (12.2.1-39) is:

$$
\delta\psi_T = \delta\psi_P - \delta\alpha \tag{12.2.1-40}
$$

where

$\delta\psi_T$, $\delta\alpha$ = Errors in the computer calculated true heading, wander angle parameters $\hat{\psi}_T$, $\hat{\alpha}$.

Equations (12.2.3-37) (in subsequent Section 12.2.3) define $\delta\alpha$ as a function of the components of $\underline{\varepsilon}^N$. Substituting $\delta\alpha$ from (12.2.3-37) and $\delta\psi_P$ from (12.2.1-38) into (12.2.1-40) obtains an expression for $\delta\psi_T$ as a function of $\underline{\gamma}^N$ and $\underline{\varepsilon}^N$ components:

$$\begin{aligned} \delta\psi_T \;=\; & \gamma_{ZN} - \tan\theta\left(\gamma_{XN}\sin\psi_P + \gamma_{YN}\cos\psi_P\right) - \varepsilon_{ZN} \\ & + \tan l\left(\varepsilon_{YN}\cos\alpha + \varepsilon_{XN}\sin\alpha\right) \end{aligned} \tag{12.2.1-41}$$

where

$\varepsilon_{XN}, \varepsilon_{YN}, \varepsilon_{ZN}$ = N Frame X, Y, Z components of $\underline{\varepsilon}^N$.

Equation (12.2.1-17) can be used to also define error relationships between the Euler angle errors and the errors in $\underline{\psi}^N$ and $\underline{\varepsilon}^N$. Equations (12.2.1-36), (12.2.1-38) and (12.2.1-41) with (12.2.1-17) become:

$$\begin{aligned} \psi_{XN} &= -\,\delta\phi\cos\theta\sin\psi_P - \delta\theta\cos\psi_P - \varepsilon_{XN} \\ \psi_{YN} &= -\,\delta\phi\cos\theta\cos\psi_P + \delta\theta\sin\psi_P - \varepsilon_{YN} \\ \psi_{ZN} &= -\,\delta\phi\sin\theta + \delta\psi_P - \varepsilon_{ZN} \end{aligned} \tag{12.2.1-42}$$

$$\begin{aligned} \delta\phi &= -\sec\theta\left[\left(\psi_{XN} + \varepsilon_{XN}\right)\sin\psi_P + \left(\psi_{YN} + \varepsilon_{YN}\right)\cos\psi_P\right] \\ \delta\theta &= \left(\psi_{YN} + \varepsilon_{YN}\right)\sin\psi_P - \left(\psi_{XN} + \varepsilon_{XN}\right)\cos\psi_P \\ \delta\psi_P &= \psi_{ZN} + \varepsilon_{ZN} - \tan\theta\left[\left(\psi_{XN} + \varepsilon_{XN}\right)\sin\psi_P + \left(\psi_{YN} + \varepsilon_{YN}\right)\cos\psi_P\right] \end{aligned} \tag{12.2.1-43}$$

$$\begin{aligned} \delta\psi_T \;=\; & \psi_{ZN} - \tan\theta\left[\left(\psi_{XN} + \varepsilon_{XN}\right)\sin\psi_P + \left(\psi_{YN} + \varepsilon_{YN}\right)\cos\psi_P\right] \\ & + \tan l\left(\varepsilon_{YN}\cos\alpha + \varepsilon_{XN}\sin\alpha\right) \end{aligned} \tag{12.2.1-44}$$

where

$\psi_{XN}, \psi_{YN}, \psi_{ZN}$ = N Frame X, Y, Z components of $\underline{\psi}^N$. Note that although the notation is similar, $\underline{\psi}^N$ and its components are not related to ψ_P, ψ_T except through the previous error equation equivalencies.

Finally, it is instructive to note that from Equations (12.2.3-19) (in subsequent Section 12.2.3), the $\varepsilon_{XN}, \varepsilon_{YN}$ terms in the previous expressions can be defined in terms of position errors as:

$$\varepsilon_{XN} = -\frac{1}{R}\,\delta R_{YN} \qquad \varepsilon_{YN} = \frac{1}{R}\,\delta R_{XN} \tag{12.2.1-45}$$

where

$\delta R_{XN}, \delta R_{YN}$ = N Frame X, Y (i.e., horizontal) components of the position error vector $\delta \underline{R}^N$.

R = Distance from earth's center to the navigation system (i.e., magnitude of the position vector from earth's center).

12.2.2 VELOCITY ERROR PARAMETERS

Velocity errors can be defined in several coordinate frames. In this section we will consider velocity errors defined in the E and N Frames, viz.:

$$\delta \underline{V}^E \equiv \hat{\underline{v}}^E - \underline{v}^E \tag{12.2.2-1}$$

$$\delta \underline{v}^N \equiv \hat{\underline{v}}^N - \underline{v}^N \tag{12.2.2-2}$$

where

$\delta \underline{V}$ = Error in velocity relative to the earth measured (defined) in the E Frame.

$\delta \underline{V}^E$ = $\delta \underline{V}$ projected on E Frame axes.

$\delta \underline{v}$ = Error in velocity relative to the earth measured (defined) in the N Frame.

$\delta \underline{v}^N$ = $\delta \underline{v}$ projected on N Frame axes.

Note from the definition of the velocity vector $\underline{v}$ in Section 4.3 (by Equation (4.3-1)) that the velocity errors defined in Equations (12.2.2-1) and (12.2.2-2) are errors in $\underline{v}$, the velocity (or position rate) relative to earth Frame E. In some applications, the concept of a velocity vector relative to the inertial I Frame is useful (denoted herein as $\underline{\upsilon}$). At the end of this section we will address the error in $\underline{\upsilon}$ and its relationship to the above defined errors in $\underline{v}$.

We now project the Equation (12.2.2-1) defined velocity error $\delta \underline{V}$ onto N Frame axes to obtain:

$$\delta \underline{V}^N = C_E^N \, \delta \underline{V}^E \tag{12.2.2-3}$$

where

$\delta\underline{V}^N$ = Velocity error defined in the E Frame (by Equation (12.2.2-1)) and projected (superscript designator) on N Frame axes.

Application of generalized Equations (3.5.4-10) defines the relationship between $\delta\underline{V}^N$ and $\delta\underline{v}^N$:

$$\delta\underline{V}^N = \delta\underline{v}^N + \underline{\beta}^N_{N\,\mathrm{to}\,E} \times \underline{v}^N \tag{12.2.2-4}$$

where

$\underline{\beta}^N_{N\,\mathrm{to}\,E}$ = Rotation angle error vector associated with $\hat{C}^E_N$ considering Frame N to be misaligned, as projected on Frame N axes.

From the definition of $\underline{\varepsilon}^N$ in Section 12.2.1, we see that it is identical to $\underline{\beta}^N_{N\,\mathrm{to}\,E}$, hence, (12.2.2-4) becomes:

$$\delta\underline{V}^N = \delta\underline{v}^N + \underline{\varepsilon}^N \times \underline{v}^N \tag{12.2.2-5}$$

We next discuss the error in the inertially (I Frame) defined velocity vector $\underline{\upsilon}$ and its relationship to the error in $\underline{v}$, the velocity vector relative to the earth fixed E Frame. Note from Section 12.1.4 (Equation (12.1.4-2)) that the formal definition for the inertial frame relative velocity vector $\underline{\upsilon}$ is the time rate of change of position in the I Frame:

$$\underline{\upsilon}^I \equiv \dot{\underline{R}}^I \tag{12.2.2-6}$$

where

$\underline{\upsilon}^I$ = Rate of change of position relative to non-rotating I Frame inertial space as projected on I Frame axes.

In contrast, the $\underline{v}$ earth relative velocity vector is defined from Equation (4.3-1) as:

$$\underline{v}^E \equiv \dot{\underline{R}}^E \tag{12.2.2-7}$$

The $\underline{\upsilon}$ and $\underline{v}$ velocity vectors are related in the E Frame according to the E Frame version of Equation (12.1.4-5):

$$\underline{\upsilon}^E = \underline{v}^E + \underline{\omega}^E_{IE} \times \underline{R}^E \tag{12.2.2-8}$$

where

$\underline{\omega}^E_{IE}$ = Angular rate of the E Frame relative to the I Frame (subscript designator) as projected on E Frame axes (superscript designator).

We now define the error in $\underline{\upsilon}$ from the I Frame relation:

$$\delta\underline{\upsilon}^I \equiv \hat{\underline{\upsilon}}^I - \underline{\upsilon}^I \qquad (12.2.2\text{-}9)$$

where

$\delta\underline{\upsilon}$ = Difference between the computed and true velocity relative to inertial space measured (defined) in the I Frame.

$\delta\underline{\upsilon}^I$ = $\delta\underline{\upsilon}$ projected on I Frame axes.

The relationship between $\delta\underline{\upsilon}$ and $\delta\underline{V}$ is developed by first rewriting (12.2.2-9) in the equivalent form:

$$\delta\underline{\upsilon}^I = \hat{C}_E^I\, \hat{\underline{\upsilon}}^E - C_E^I\, \underline{\upsilon}^E \qquad (12.2.2\text{-}10)$$

where

C_E^I = Direction cosine matrix that transforms vectors from the E Frame to the I Frame.

But from Equation (12.2.1-20):

$$\hat{C}_E^I = C_E^I \qquad (12.2.2\text{-}11)$$

Because $\underline{\omega}_{IE}^E$ is constant (See Equations (12.1.3-7)) we can also write:

$$\hat{\underline{\omega}}_{IE}^E = \underline{\omega}_{IE}^E \qquad (12.2.2\text{-}12)$$

Substituting (12.2.2-8), (12.2.2-11) and (12.2.2-12) into (12.2.2-10) yields:

$$\begin{aligned}\delta\underline{\upsilon}^I &= C_E^I\left(\hat{\underline{v}}^E + \hat{\underline{\omega}}_{IE}^E \times \hat{\underline{R}}^E - \underline{v}^E - \underline{\omega}_{IE}^E \times \underline{R}^E\right) \\ &= C_E^I\left(\hat{\underline{v}}^E - \underline{v}^E + \underline{\omega}_{IE}^E \times \left(\hat{\underline{R}}^E - \underline{R}^E\right)\right)\end{aligned} \qquad (12.2.2\text{-}13)$$

The error in the system computed position vector $\hat{\underline{R}}$ is defined in the E Frame as:

$$\delta\underline{R}^E \equiv \hat{\underline{R}}^E - \underline{R}^E \qquad (12.2.2\text{-}14)$$

where

$\delta\underline{R}^E$ = Error in $\hat{\underline{R}}^E$.

Applying (12.2.2-1) and (12.2.2-14) to (12.2.2-13) then obtains:

$$\delta \underline{\upsilon}^{I} = \delta \underline{V}^{I} + \underline{\omega}_{IE}^{I} \times \delta \underline{R}^{I} \tag{12.2.2-15}$$

In the N Frame Equation (12.2.2-15) has the equivalent final form:

$$\delta \underline{\upsilon}^{N} = \delta \underline{V}^{N} + \underline{\omega}_{IE}^{N} \times \delta \underline{R}^{N} \tag{12.2.2-16}$$

Equation (12.2.2-16) shows the relationship between the inertial and earth frame defined velocity vector errors.

We conclude this section with a discussion of a velocity error $\delta \underline{\mathcal{V}}$ defined in a locally level geographic frame and its relationship with the E Frame defined velocity error $\delta \underline{V}$ and the N Frame defined velocity error $\delta \underline{v}$. First, we define the geographic frame velocity error as:

$$\delta \underline{\mathcal{V}}^{Geo} \equiv \hat{\underline{v}}^{Geo} - \underline{v}^{Geo} \tag{12.2.2-17}$$

where

Geo = Locally level geographic coordinate frame defined with its Z axis upward along the local geodetic vertical, Y axis north (and horizontal) with X axis east (and horizontal).

$\delta \underline{\mathcal{V}}$ = Difference between the computed and true velocity relative to the earth as measured (defined) in the Geo Frame.

$\delta \underline{\mathcal{V}}^{Geo}$ = $\delta \underline{\mathcal{V}}$ projected on Geo Frame axes.

The velocity vector relative to the earth $\underline{v}$ has components in the E and Geo Frames related by:

$$\underline{v}^{E} = C_{Geo}^{E}\, \underline{v}^{Geo} \tag{12.2.2-18}$$

Application of generalized Equation (3.5.4-10) (rearranged) to (12.2.2-18) provides the relationship between velocity error $\delta \underline{\mathcal{V}}$ defined in the Geo Frame (i.e., Equation (12.2.2-17)) and velocity error $\delta \underline{V}$ defined in the E Frame (by Equation (12.2.2-1)):

$$\delta \underline{\mathcal{V}}^{Geo} = \delta \underline{V}^{Geo} - \underline{\vartheta}^{Geo} \times \underline{v}^{Geo} \tag{12.2.2-19}$$

where

$\underline{\vartheta}^{Geo}$ = Rotation angle error associated with $\hat{C}_{Geo}^{E}$ (considering the Geo Frame to be misaligned) as projected on Geo Frame axes.

The $\delta \underline{V}^{Geo}$ term for (12.2.2-19) can be calculated from the N Frame $\delta \underline{V}$ components, viz.:

$$\delta\underline{V}^{Geo} = C_N^{Geo}\,\delta\underline{V}^N \tag{12.2.2-20}$$

The C_N^{Geo} matrix in (12.2.2-20) is a transformation through the wander angle α as shown by the transpose of (6.1.3-8):

$$C_N^{Geo} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{12.2.2-21}$$

where

α = Wander angle defined as the angle between the N frame Y axis and true North, measured positive around the local upward geodetic vertical (the Z axis of the N Frame).

From their definitions in Section 2.2, the Geo Frame is equivalent to the N Frame with a zero wander angle. Consequently, the C_{Geo}^E matrix can be expressed in terms of latitude (l), longitude (L) Euler angles as in Section 4.4.2.1 with the wander angle α set to zero. Then the $\underline{\vartheta}^{Geo}$ angle error vector in (12.2.2-19) can be equated to the latitude, longitude (l, L) Euler angle error terms using generalized Equations (3.5.3-31) and Section 4.4.2.1 for definition of the l, L Euler rotation angles and axes:

$$\underline{\vartheta}^{Geo} = \delta L\, \underline{u}_{YE}^{Geo} - \delta l\, \underline{u}_{XGeo}^{Geo} \tag{12.2.2-22}$$

where

$\underline{u}_{YE}^{Geo}$ = Unit vector along the earth polar axis (the E Frame Y axis) projected on Geo Frame axes (corresponding to the $\underline{u}_{\zeta A}$ axis in (3.5.3-31)).

$\underline{u}_{XGeo}^{Geo}$ = Unit vector along the local horizontal east direction (the Geo Frame X axis) projected on Geo Frame axes. $\underline{u}_{XGeo}^{Geo}$ corresponds with the $\underline{u}_{\zeta A_1}$ axis in (3.5.3-31).

Note the negative sign for the δl term in (12.2.2-22) because latitude is defined in Section 4.4.2.1 as a negative Euler angle rotation.

We equate $\underline{u}_{YE}^{Geo}$ in (12.2.2-22) to the sum of its vertical and horizontal components; the vertical component equals the cosine of the angle between the local vertical and the E Frame earth polar Y axis (i.e., $u_{Up_{YE}}$ in Section 5.2.3 defined in Equation (5.2.3-1) to equal the sine of geodetic latitude); the horizontal component lies north along the Geo Frame Y axis with magnitude equal to the complement of the vertical component (i.e., cos l):

$$\underline{u}_{YE}^{Geo} = \cos l\ \underline{u}_{YGeo}^{Geo} + \sin l\ \underline{u}_{ZGeo}^{Geo} \tag{12.2.2-23}$$

The latitude, longitude errors in (12.2.2-22) can be expressed in terms of errors in the C_N^E matrix beginning with generalized Equation (3.5.3-31) transformed to the Geo Frame, and Section 4.4.2.1 for definition of the L, l , α (longitude, latitude, wander angle) Euler rotations and axes:

$$\underline{\varepsilon}^{Geo} = \delta L\ \underline{u}_{YE}^{Geo} - \delta l\ \underline{u}_{XGeo}^{Geo} + \delta\alpha\ \underline{u}_{ZGeo}^{Geo} \tag{12.2.2-24}$$

where

$\underline{\varepsilon}^{Geo}$ = Rotation angle error associated with $\widehat{C}_N^E$ (considering the N Frame to be misaligned) projected on Geo Frame axes.

$\delta\alpha$ = Error in the wander angle α.

$\underline{u}_{ZGeo}^{Geo}$ = Unit vector along the Geo and N Frame Z axes (corresponding to the $\underline{u}_{\zeta A_2}$ axis in (3.5.3-31)).

Substituting (12.2.2-23) into (12.2.2-24) finds after rearrangement:

$$\underline{\varepsilon}^{Geo} = - \delta l\ \underline{u}_{XGeo}^{Geo} + \delta L \cos l\ \underline{u}_{YGeo}^{Geo} + \left(\delta\alpha + \delta L \sin l\right) \underline{u}_{ZGeo}^{Geo} \tag{12.2.2-25}$$

Latitude, longitude are then obtained from the dot product of (12.2.2-25) with $\underline{u}_{XGeo}^{Geo}$, $\underline{u}_{YGeo}^{Geo}$ and rearrangement:

$$\delta l = - \underline{u}_{XGeo}^{Geo} \cdot \underline{\varepsilon}_H^{Geo} \qquad \delta L = \underline{u}_{YGeo}^{Geo} \cdot \underline{\varepsilon}_H^{Geo} \sec l \tag{12.2.2-26}$$

where

$\underline{\varepsilon}_H^{Geo}$ = Horizontal component of $\underline{\varepsilon}^{Geo}$. The $\underline{\varepsilon}_H^{Geo}$ component of $\underline{\varepsilon}^{Geo}$ can be used in (12.2.2-26) because the dot product of the vertical $\underline{\varepsilon}^{Geo}$ component with $\underline{u}_{XGeo}^{Geo}$ and $\underline{u}_{YGeo}^{Geo}$ is zero.

The $\underline{\varepsilon}_H^{Geo}$ term for (12.2.2-26) can be calculated from the N Frame horizontal $\underline{\varepsilon}$ components, viz.:

$$\underline{\varepsilon}_H^{Geo} = C_N^{Geo}\ \underline{\varepsilon}_H^N \tag{12.2.2-27}$$

With (12.2.2-26) and (12.2.2-23), Equation (12.2.2-22) for $\underline{\vartheta}^{Geo}$ becomes:

$$\begin{aligned}\underline{\vartheta}^{Geo} &= \sec l \left(\underline{u}_{YGeo}^{Geo} \cdot \underline{\varepsilon}_{H}^{Geo}\right) \underline{u}_{YE}^{Geo} + \left(\underline{u}_{XGeo}^{Geo} \cdot \underline{\varepsilon}_{H}^{Geo}\right) \underline{u}_{XGeo}^{Geo} \\ &= \varepsilon_{XGeo}\, \underline{u}_{XGeo}^{Geo} + \varepsilon_{YGeo}\, \underline{u}_{YGeo}^{Geo} + \varepsilon_{YGeo} \tan l\, \underline{u}_{ZGeo}^{Geo}\end{aligned} \tag{12.2.2-28}$$

where

$\varepsilon_{XGeo}, \varepsilon_{YGeo}$ = X, Y components of $\underline{\varepsilon}_{H}^{Geo}$.

The $\underline{\varepsilon}_{H}^{Geo}$ term in (12.2.2-28) can also be expressed in terms of horizontal position error (defined in (12.2.2-14)) using the horizontal component of Equation (12.2.3-19) (in the next section) in Geo Frame coordinates. Then (12.2.2-28) with (12.2.3-19) becomes:

$$\underline{\vartheta}^{Geo} = -\frac{1}{R}\,\delta R_{YGeo}\, \underline{u}_{XGeo}^{Geo} + \frac{1}{R}\,\delta R_{XGeo}\, \underline{u}_{YGeo}^{Geo} + \frac{1}{R}\,\delta R_{XGeo} \tan l\, \underline{u}_{ZGeo}^{Geo} \tag{12.2.2-29}$$

where

$\delta R_{XGeo}, \delta R_{YGeo}$ = X, Y components of $\delta\underline{R}_{H}^{Geo}$, the horizontal components of the Equation (12.2.2-14) defined $\delta\underline{R}$ position error vector, as projected on Geo Frame axes.

The $\delta\underline{R}_{H}^{Geo}$ components for (12.2.2-29) can be calculated from the N Frame horizontal $\delta\underline{R}$ components via:

$$\delta\underline{R}_{H}^{Geo} = C_{N}^{Geo}\, \delta\underline{R}_{H}^{N} \tag{12.2.2-30}$$

with C_{N}^{Geo} provided by (12.2.2-21).

Summarizing for a moment, the previous development has shown that $\delta\underline{\mathcal{V}}^{Geo}$ can be expressed as a function of $\delta\underline{V}^{Geo}$ using (12.2.2-19) with $\underline{\vartheta}^{Geo}$ calculated from horizontal $\underline{\varepsilon}$ components (using (12.2.2.-28)) or from horizontal $\delta\underline{R}$ components (using (12.2.2-29)). The Geo Frame horizontal components of $\delta\underline{V}$, $\underline{\varepsilon}$ and $\delta\underline{R}$ for (12.2.2-19), (12.2.2-28) and (12.2.2-29) can be calculated from the N Frame components using (12.2.2-20), (12.2.2-21), (12.2.2-27) and (12.2.2-30).

We can also develop an equivalent set of expressions for $\delta\underline{\mathcal{V}}^{Geo}$ as a function of the Geo Frame components of $\delta\underline{v}$. Substituting the Geo Frame form of Equation (12.2.2-5) into (12.2.2-19) yields:

$$\delta\underline{\mathcal{V}}^{Geo} = \delta\underline{v}^{Geo} + \left(\underline{\varepsilon}^{Geo} - \underline{\vartheta}^{Geo}\right) \times \underline{v}^{Geo} \tag{12.2.2-31}$$

Using the difference between (12.2.2-24) and (12.2.2-22) for $\left(\underline{\varepsilon}^{Geo} - \underline{\vartheta}^{Geo}\right)$ in (12.2.2-31) then yields the desired form:

$$\delta\underline{\mathcal{V}}^{Geo} = \delta\underline{v}^{Geo} + \delta\alpha \, \underline{u}_{ZGeo}^{Geo} \times \underline{v}^{Geo} \tag{12.2.2-32}$$

Equations (12.2.3-37) (in the next section) provides an expression relating the wander angle error in (12.2.2-32) to the N Frame components of $\underline{\varepsilon}$. The $\delta\underline{v}^{Geo}$ term in (12.2.2-32) can be calculated from the N Frame components using:

$$\delta\underline{v}^{Geo} = C_N^{Geo} \, \delta\underline{v}^N \tag{12.2.2-33}$$

with C_N^{Geo} provided by (12.2.2-21).

12.2.3 POSITION ERROR PARAMETERS

The position error in an inertial navigation system can be defined as the error in the position vector that describes the system position location relative to the earth:

$$\delta\underline{R}^E \equiv \hat{\underline{R}}^E - \underline{R}^E \tag{12.2.3-1}$$

where

$\underline{R}^E$ = Position vector from earth's center to the navigation system as described in E Frame axes.

$\hat{\underline{R}}^E$ = Value for $\underline{R}^E$ calculated in the navigation system computer.

$\delta\underline{R}^E$ = Error in $\hat{\underline{R}}^E$ as projected (superscript) on E Frame axes.

We can also describe $\delta\underline{R}^E$ in the N Frame:

$$\delta\underline{R}^N \equiv C_E^N \, \delta\underline{R}^E \tag{12.2.3-2}$$

where

$\delta \underline{R}^N$ = Error in $\hat{\underline{R}}^E$ as projected on N Frame axes.

Alternatively, as discussed in Section 4.4, navigation system position location relative to the earth can be described by the angular orientation of the N Frame relative to the E Frame (e.g., latitude, longitude) and the altitude above the surface of the earth. Using the C_N^E direction cosine matrix to represent the N to E relative angular orientation leads (with Equations (12.2.1-11) and (12.2.1-12)) to the following associated position errors:

$$\delta C_N^E = C_N^E \left(\underline{\varepsilon}^N \times\right) \tag{12.2.3-3}$$

$$\left(\underline{\varepsilon}^N \times\right) = \left(C_N^E\right)^T \hat{C}_N^E - I \tag{12.2.3-4}$$

$$\delta h = \hat{h} - h \tag{12.2.3-5}$$

where

$\underline{\varepsilon}^N$ = Rotation angle error vector associated with the $\hat{C}_N^E$ matrix considering the N Frame to be misaligned, as projected on Frame N axes.

δh = Error in the system computed altitude $\hat{h}$.

Equivalencies between the $\delta \underline{R}^N$ and $\underline{\varepsilon}^N$, δh position error parameters can developed from the following approximate form:

$$\underline{R}^E \approx R\, \underline{u}_{ZN}^E \tag{12.2.3-6}$$

where

$\underline{u}_{ZN}^E$ = Unit vector upward along the local geodetic vertical (i.e., along the N Frame Z axis) as projected on E Frame axes.

R = Distance from earth's center to the navigation system (i.e., the magnitude of $\underline{R}^E$).

Equation (12.2.3-6) approximates earth's shape as a sphere rather than an ellipsoid of revolution (with ellipticity e). For error analysis purposes this is generally an insignificant error. Taking the differential of (12.2.3-6) then obtains:

$$\delta \underline{R}^E = \delta R\, \underline{u}_{ZN}^E + R\, \delta \underline{u}_{ZN}^E \tag{12.2.3-7}$$

with

$$\delta R \equiv \hat{R} - R \tag{12.2.3-8}$$

where

δR = Error in system computed value for R.

$\delta \underline{u}_{ZN}^{E}$ = Error in system computer version of $\underline{u}_{ZN}^{E}$.

We can also write:

$$\underline{u}_{ZN}^{E} = C_{N}^{E}\, \underline{u}_{ZN}^{N} \tag{12.2.3-9}$$

where

$\underline{u}_{ZN}^{N}$ = Unit vector along the N Frame Z-axis as projected on N Frame axes.

Since $\underline{u}_{ZN}^{N}$ is known explicitly by definition (i.e., components of 0, 0, 1), the associated error in the system computer version is zero, and the differential of (12.2.3-9) is:

$$\delta \underline{u}_{ZN}^{E} = \delta C_{N}^{E}\, \underline{u}_{ZN}^{N} \tag{12.2.3-10}$$

Substituting Equation (12.2.3-3) into (12.2.3-10) then yields for $\delta \underline{u}_{ZN}^{E}$ in (12.2.3-7):

$$\delta \underline{u}_{ZN}^{E} = C_{N}^{E} \left(\underline{\varepsilon}^{N} \times \right) \underline{u}_{ZN}^{N} = C_{N}^{E} \left(\underline{\varepsilon}^{N} \times \underline{u}_{ZN}^{N} \right) \tag{12.2.3-11}$$

The δR for Equation (12.2.3-7) is obtained from the R definition as in generalized Equations (3.1.1-4) - (3.1.1-5) (with $\underline{W} = \underline{V}$):

$$R^{2} = \underline{R}^{E} \cdot \underline{R}^{E} \tag{12.2.3-12}$$

Taking the differential of (12.2.3-12) yields:

$$2\, R\, \delta R = 2\, \underline{R}^{E} \cdot \delta \underline{R}^{E} \tag{12.2.3-13}$$

Further refinement of Equation (12.2.3-13) is achieved by substituting (12.2.3-6) for $\underline{R}^{E}$:

$$\delta R = \underline{u}_{ZN}^{E} \cdot \delta \underline{R}^{E} \tag{12.2.3-14}$$

We now substitute (12.2.3-11) for $\delta \underline{u}_{ZN}^{E}$ into Equation (12.2.3-7) to find:

$$\delta \underline{R}^{E} = \delta R\, \underline{u}_{ZN}^{E} + R\, C_{N}^{E} \left(\underline{\varepsilon}^{N} \times \underline{u}_{ZN}^{N} \right) \tag{12.2.3-15}$$

or, after transforming to the N Frame:

$$\delta\underline{R}^N = R\left(\underline{\varepsilon}^N \times \underline{u}_{ZN}^N\right) + \delta R\, \underline{u}_{ZN}^N \tag{12.2.3-16}$$

Equation (12.2.3-16) defines the $\delta\underline{R}^N$ position error vector in terms of the equivalent $\underline{\varepsilon}^N$, δR position error parameters.

The converse of (12.2.3-16) is obtained by taking the cross-product with $\underline{u}_{ZN}^N$:

$$\underline{u}_{ZN}^N \times \delta\underline{R}^N = R\, \underline{u}_{ZN}^N \times \left(\underline{\varepsilon}^N \times \underline{u}_{ZN}^N\right) \tag{12.2.3-17}$$

To reduce (12.2.3-17) we make use of the Equation (3.1.1-16) vector triple product identity with which Equation (12.2.3-17) becomes:

$$\underline{u}_{ZN}^N \times \delta\underline{R}^N = R\left[\underline{\varepsilon}^N - \underline{u}_{ZN}^N\left(\underline{u}_{ZN}^N \cdot \underline{\varepsilon}^N\right)\right] \tag{12.2.3-18}$$

Identifying the $\underline{u}_{ZN}^N \cdot \underline{\varepsilon}^N$ term in (12.2.3-18) as the Z (upward vertical) component of $\underline{\varepsilon}^N$ and using (12.2.3-14) in the N Frame then, after rearrangement, allows us to define the $\underline{\varepsilon}^N$, δR position error parameters in terms of $\delta\underline{R}^N$:

$$\underline{\varepsilon}^N = \frac{1}{R}\left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) + \varepsilon_{ZN}\, \underline{u}_{ZN}^N \tag{12.2.3-19}$$

$$\delta R = \underline{u}_{ZN}^N \cdot \delta\underline{R}^N \tag{12.2.3-20}$$

where

ε_{ZN} = Z component of $\underline{\varepsilon}^N$.

Note in Equation (12.2.3-19), that because $\underline{u}_{ZN}^N$ is along the Z axis of the N Frame, its cross-product with $\delta\underline{R}^N$ produces zero contribution to $\underline{\varepsilon}^N$ from the $\delta\underline{R}^N$ vertical component (δR). However, the ε_{ZN} vertical component must be provided from another source. Similarly, in Equation (12.2.3-16), the vertical ε_{ZN} component of $\underline{\varepsilon}^N$ has no effect on $\delta\underline{R}^N$, however, the vertical δR component of $\delta\underline{R}^N$ must be provided from another source.

The equivalent to Equations (12.2.3-16), (12.2.3-19) and (12.2.3-20) can also be obtained for $\delta\underline{R}^N$ in terms of $\underline{\varepsilon}^N$ / δh, and for $\underline{\varepsilon}^N$ / δh in terms $\delta\underline{R}^N$ / ε_{ZN}, by using the combined R, R_S and $u_{ZN_{YE}}$ expressions from Equations (12.1.2-6):

$$R = \left(1 - D_{23}^2\, e\right) R_0 + h \tag{12.2.3-21}$$

where

$$D_{23} = \text{Element of } C_N^E \text{ in row 2, column 3.}$$

The differential of (12.2.3-21) after rearrangement is:

$$\delta R = \delta h - 2\, e\, D_{23}\, \delta D_{23}\, R_0 \qquad (12.2.3\text{-}22)$$

Equation (12.2.1-11) shows that δD_{23} in (12.2.3-22) (an element of δC_N^E) is on the order of $\underline{\varepsilon}^N$. Equation (12.2.3-19) shows that $\underline{\varepsilon}^N$ is on the order of $\delta \underline{R}^N / R$. Hence, $\delta D_{23}\, R_0$ in Equation (12.2.3-22) is on the order of $(R_0 / R)\ \delta \underline{R}^N$. Since this term multiplies e in Equation (12.2.3-22), we can conclude that it is negligible compared to the leading δR term. Thus, Equation (12.2.3-22) simplifies to the approximate form:

$$\delta R \approx \delta h \qquad (12.2.3\text{-}23)$$

With (12.2.3-23), Equations (12.2.3-16), (12.2.3-19) and (12.2.3-20) become the equivalent forms:

$$\delta \underline{R}^N = R \left(\underline{\varepsilon}^N \times \underline{u}_{ZN}^N \right) + \delta h\, \underline{u}_{ZN}^N \qquad (12.2.3\text{-}24)$$

and the converse:

$$\underline{\varepsilon}^N = \frac{1}{R} \left(\underline{u}_{ZN}^N \times \delta \underline{R}^N \right) + \varepsilon_{ZN}\, \underline{u}_{ZN}^N \qquad (12.2.3\text{-}25)$$

$$\delta h = \underline{u}_{ZN}^N \cdot \delta \underline{R}^N \qquad (12.2.3\text{-}26)$$

Equations (12.2.3-24) - (12.2.3-26) define $\delta \underline{R}^N$ in terms of $\underline{\varepsilon}^N$, δh, and define $\underline{\varepsilon}^N$, δh in terms $\delta \underline{R}^N$, ε_{ZN}.

For error analysis purposes, we will at times make use of error parameters defined in the I Frame. We show here, that the $\delta \underline{R}$ position error parameter defined in the E Frame by (12.2.3-1) is identical if defined in the I Frame. To do this, we temporarily identify the position error defined in the I Frame as:

$$\delta \underline{r}^I \equiv \hat{\underline{R}}^I - \underline{R}^I \qquad (12.2.3\text{-}27)$$

where

$$\delta \underline{r}^I = \text{Error in } \hat{\underline{R}}^I \text{ as evaluated in the I Frame.}$$

The $\delta\underline{r}$ position error in the E Frame is with (12.2.3-27):

$$\delta\underline{r}^E = C_I^E \, \delta\underline{r}^I = C_I^E \, \hat{\underline{R}}^I - C_I^E \, \underline{R}^I = C_I^E \, \hat{\underline{R}}^I - \underline{R}^E \tag{12.2.3-28}$$

where

C_I^E = Direction cosine matrix that transforms vectors from the I Frame to the E Frame.

But from Equation (12.2.1-20):

$$C_I^E = \hat{C}_I^E \tag{12.2.3-29}$$

Substituting (12.2.3-29) in (12.2.3-28) then obtains:

$$\delta\underline{r}^E = \hat{C}_I^E \, \hat{\underline{R}}^I - \underline{R}^E = \hat{\underline{R}}^E - \underline{R}^E \tag{12.2.3-30}$$

Comparing (12.2.3-30) with (12.2.3-1) we see then as stipulated that:

$$\delta\underline{r}^E = \delta\underline{R}^E \tag{12.2.3-31}$$

We can also relate the $\underline{\varepsilon}^N$ and $\delta\underline{R}^N$ position error parameters to errors in the Euler angles typically used to describe the C_N^E matrix for navigation purposes (i.e., from Section 4.4.2.1, latitude, longitude and wander angle). This is achieved analytically by first defining the C_N^E matrix in terms of the intermediate direction cosine matrices that constitute the Euler angle rotations. Applying general Equation (3.2.3.1-1) we have:

$$C_N^E = C_{E_1}^E \, C_{E_2}^{E_1} \, C_N^{E_2} \tag{12.2.3-32}$$

where, from Section 4.4.2.1:

E_1 Frame = Frame E (as defined in Section 2.2), but after rotating it about E Frame axis Y through the longitude Euler angle.

E_2 Frame = Frame E_1 after rotating it negatively about E_1 Frame axis X through the geodetic latitude Euler angle.

N Frame = Navigation Frame (as defined in Section 2.2) obtained by rotating Frame E_2 about the E_2 Frame Z-axis through the wander angle Euler rotation.

From the previous coordinate frame definitions and use of generalized Equations (3.2.3-3) - (3.2.3-4) as a guide, we can also write:

$$C_{E_1}^{E} = \begin{bmatrix} \cos L & 0 & \sin L \\ 0 & 1 & 0 \\ -\sin L & 0 & \cos L \end{bmatrix} \qquad C_{E_2}^{E_1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos l & \sin l \\ 0 & -\sin l & \cos l \end{bmatrix}$$

$$C_{N}^{E_2} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{12.2.3-33}$$

$$\underline{u}_{\zeta E}^{E} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \underline{u}_{\zeta E_1}^{E_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \underline{u}_{\zeta E_2}^{E_2} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

where

$\underline{u}_{\zeta E}^{E}, \underline{u}_{\zeta E_1}^{E_1}, \underline{u}_{\zeta E_2}^{E_2}$ = Unit vectors along the E Frame Y-axis, E_1 Frame X-axis and E_2 Frame Z-axis.

L = Longitude measured around $\underline{u}_{\zeta E}^{E}$ from the Greenwich meridian and positive for positive rotations around $\underline{u}_{\zeta E}^{E}$.

l = Geodetic latitude measured around $\underline{u}_{\zeta E_1}^{E_1}$ and positive for negative rotations around $\underline{u}_{\zeta E_1}^{E_1}$.

α = Wander angle measured around $\underline{u}_{\zeta E_2}^{E_2}$ and positive for positive rotations around $\underline{u}_{\zeta E_2}^{E_2}$.

Note that the form of the $C_{E_2}^{E_1}$ matrix in Equations (12.2.3-33) has an inverse sign configuration from the generalized X-axis rotation matrix in Equations (3.2.3-4) (i.e., the ϕ expression) because the latitude Euler angle l is defined to be a negative rotation about the E_1 Frame X-axis.

We now apply generalized Equations (3.5.3-31) - (3.5.3-32) to obtain an expression for the Equation (12.2.1-12) defined $\underline{\varepsilon}^{N}$ error in C_{N}^{E} as a function of the errors in the associated Euler angles:

$$\underline{\varepsilon}^{N} = \delta L \, \underline{u}_{\zeta E}^{N} - \delta l \, \underline{u}_{\zeta E_1}^{N} + \delta\alpha \, \underline{u}_{\zeta E_2}^{N}$$

$$\underline{u}_{\zeta E}^{N} = C_{E_2}^{N} C_{E_1}^{E_2} \underline{u}_{\zeta E}^{E} \qquad \underline{u}_{\zeta E_1}^{N} = C_{E_2}^{N} \underline{u}_{\zeta E_1}^{E_1} \tag{12.2.3-34}$$

Note that the sign of the δl term is negative because the l Euler rotation has been defined to be negative about axis $\underline{u}_{\zeta E_1}^{E_1}$.

Using Section 3.2.3.3, Equations (12.2.3-34) with (12.2.3-33) can then be formatted into the Figure 12.2.3-1 "Method of Least Work" diagram:

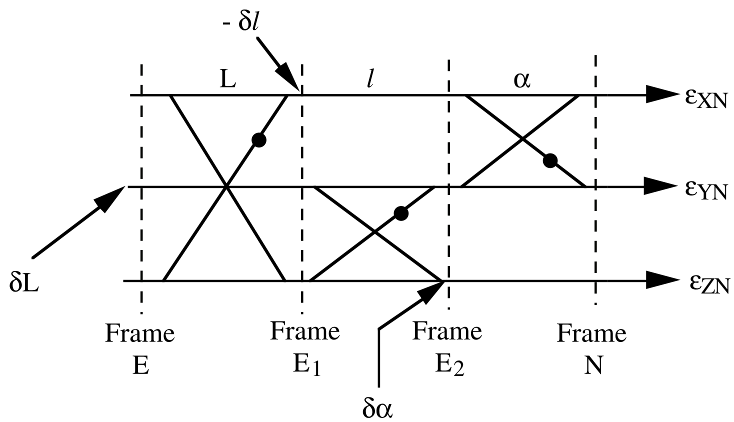


Figure 12.2.3-1 Position Rotation Angle Error Vector As A Function Of Latitude, Longitude, Wander Angle Errors

Direct reading of Figure 12.2.3-1 yields for the components of $\underline{\varepsilon}^N$:

$$\begin{aligned} \varepsilon_{XN} &= \delta L \cos l \sin \alpha - \delta l \cos \alpha \\ \varepsilon_{YN} &= \delta L \cos l \cos \alpha + \delta l \sin \alpha \\ \varepsilon_{ZN} &= \delta L \sin l + \delta \alpha \end{aligned} \tag{12.2.3-35}$$

The inverse of Equations (12.2.3-35) is obtained by first equating the X, Y Frame E_2 node inputs from the left to the inputs from the right obtained by reversing the arrow directions on the right:

$$\begin{aligned} \varepsilon_{XN} \cos \alpha - \varepsilon_{YN} \sin \alpha &= - \delta l \\ \varepsilon_{YN} \cos \alpha + \varepsilon_{XN} \sin \alpha &= \delta L \cos l \end{aligned} \tag{12.2.3-36}$$

The simple inversion of Equations (12.2.3-36) with substitution and rearrangement of the Equation (12.2.3-35) ε_{ZN} expression then provides the desired inverse relationships:

$$\delta L = \sec l \left(\varepsilon_{YN} \cos \alpha + \varepsilon_{XN} \sin \alpha\right)$$
$$\delta l = \varepsilon_{YN} \sin \alpha - \varepsilon_{XN} \cos \alpha \qquad (12.2.3\text{-}37)$$
$$\delta \alpha = \varepsilon_{ZN} - \tan l \left(\varepsilon_{YN} \cos \alpha + \varepsilon_{XN} \sin \alpha\right)$$

The equivalent relationships between latitude, longitude, wander angle errors and the $\delta \underline{R}^N$ position error vector components is determined from the X, Y components of Equation (12.2.3-19):

$$\varepsilon_{XN} = - \frac{1}{R} \delta R_{YN} \qquad \varepsilon_{YN} = \frac{1}{R} \delta R_{XN} \qquad (12.2.3\text{-}38)$$

where

$\delta R_{XN}, \delta R_{YN}$ = X, Y components of $\delta \underline{R}^N$.

Substituting Equations (12.2.3-38) into (12.2.3-35) and (12.2.3-37) then gives:

$$\delta R_{XN} = R \left(\delta L \cos l \cos \alpha + \delta l \sin \alpha\right)$$
$$\delta R_{YN} = - R \left(\delta L \cos l \sin \alpha - \delta l \cos \alpha\right) \qquad (12.2.3\text{-}39)$$
$$\delta R_{ZN} = \text{Not related to } \delta L,\ \delta l,\ \delta\alpha.$$

$$\delta L = \frac{1}{R} \sec l \left(\delta R_{XN} \cos \alpha - \delta R_{YN} \sin \alpha\right)$$
$$\delta l = \frac{1}{R} \left(\delta R_{XN} \sin \alpha + \delta R_{YN} \cos \alpha\right) \qquad (12.2.3\text{-}40)$$
$$\delta \alpha = \varepsilon_{ZN} - \frac{1}{R} \tan l \left(\delta R_{XN} \cos \alpha - \delta R_{YN} \sin \alpha\right)$$

where

δR_{ZN} = Z (upward vertical) component of $\delta \underline{R}^N$.

12.2.4 GRAVITY AND TRANSPORT RATE ERRORS

In this section we develop error expressions for the $\underline{g}_P^N$, $\underline{g}_P^E$ and $\underline{g}_P^I$ plumb-bob gravity terms in Equations (12.1.2-6), (12.1.3-7), and (12.1.4-18), and for the $\underline{\omega}_{EN}^N$ transport rate term in Equations (12.1.2-6). These will be used in subsequent sections when developing the differential equations for the navigation error parameters.

Beginning with the $\underline{g}_P^N$ expression in (12.1.2-6), we note (as discussed in Section 12.1.1) that the $\partial g_{P_{Up}}$ and $\partial g_{P_{North}}$ terms are first order compared to the dominant zero order - H(R) $\underline{u}_{ZN}^N$ leading term, hence, for error analysis purposes, the $\partial g_{P_{Up}}$ and $\partial g_{P_{North}}$ terms can generally be neglected. In addition, we recognize that the $\underline{g}_P^N$ expression in (12.1.2-6) is a truncated version with linearization of the complete set of gravity equations provided by Equations (12.1-23) - (12.1-28) and, furthermore, that the previous "complete" equation set is only an approximation of earth's actual gravitational acceleration with its local anomalies created by density and surface shape irregularities. Based on the these considerations, and noting from Equations (12.1.2-6) that $\underline{u}_{ZN}^N$ is constant, we write the N Frame plumb-bob gravity error as the differential of the (12.1.2-6) $\underline{g}_P^N$ expression, and substitute H(R) from (12.1.2-6) in the result:

$$\delta\underline{g}_P^N = -\frac{d\,H(R)}{dR}\,\underline{u}_{ZN}^N\,\delta R + \delta\underline{g}_{Mdl}^N$$
$$\frac{d\,H(R)}{dR} = -2\,\frac{\mu}{R^3} = -2\,\frac{H(R)}{R} \qquad \text{For } h \geq 0 \qquad (12.2.4\text{-}1)$$
$$\frac{d\,H(R)}{dR} = \frac{\mu}{R_S^3} = \frac{H(R)}{R} \qquad \text{For } h < 0$$

where

$\delta\underline{g}_P^N$ = Error in $\underline{g}_P^N$.

$\delta\underline{g}_{Mdl}^N$ = Modeling error in $\underline{g}_P^N$ produced by variations in Equations (12.1-23) - (12.1-28) from the true earth model, variations in the Equations (12.1.2-6) gravity expression from Equations (12.1-23) - (12.1-28), and neglecting $\partial g_{P_{Up}}$ and $\partial g_{P_{North}}$ in Equations (12.1.2-6) as previously justified.

From the Equation (12.1.2-6) $\underline{g}_P^N$ expression, H(R) is approximately equal to the magnitude of gravity (Note that the difference between plumb-bob gravity and mass attraction gravity is in the $\partial g_{P_{Up}}$ and $\partial g_{P_{North}}$ terms being neglected). Hence, with (12.2.3-23), Equation (12.2.4-1) can be written in the equivalent form:

$$\delta\underline{g}_P^N \approx F(h)\,\frac{g}{R}\,\underline{u}_{ZN}^N\,\delta h + \delta\underline{g}_{Mdl}^N$$
$$F(h) = 2 \quad \text{For } h \geq 0 \qquad F(h) = -1 \quad \text{For } h < 0 \qquad (12.2.4\text{-}2)$$

where

g = Magnitude of gravity at position $\underline{R}^N$.

$F(h)$ = Gravity error coupling parameter that characterizes the fundamental difference between gravity error values above and below earth's geoid surface.

The error in $\underline{g}_P^E$ is obtained from the differential of the $\underline{g}_P^E$ expression in Equations (12.1.3-7) using the same approximations discussed above for $\underline{g}_P^N$ and dropping the e term as negligible (because e is small):

$$\delta\underline{g}_P^E \approx - \frac{d\,H(R)}{dR}\,\frac{1}{R}\,\underline{R}^E\,\delta R + H(R)\,\frac{1}{R^2}\,\underline{R}^E\,\delta R - H(R)\,\frac{1}{R}\,\delta\underline{R}^E + \delta\underline{g}_{Mdl}^E \tag{12.2.4-3}$$

where

$\delta\underline{g}_P^E$ = Error in $\underline{g}_P^E$.

$\delta\underline{g}_{Mdl}^E$ = Modeling error in $\underline{g}_P^E$ produced by variations in Equations (12.1-23) - (12.1-28) from the true earth model, variations in the Equations (12.1.3-7) gravity expression from Equations (12.1-23) - (12.1-28), and neglecting $\partial g_{P_{Up}}$ and $\partial g_{P_{North}}$ in Equations (12.1.3-7).

Using the expressions for $\frac{d\,H(R)}{dR}$ from Equations (12.2.4-1) and substituting g for H(R), Equation (12.2.4-3) becomes after combining like terms:

$$\delta\underline{g}_P^E = F(h)\,\frac{g}{R^2}\,\underline{R}^E\,\delta R - \frac{g}{R}\left(\delta\underline{R}^E - \frac{1}{R}\,\underline{R}^E\,\delta R\right) + \delta\underline{g}_{Mdl}^E \tag{12.2.4-4}$$

with F(h) as defined in Equations (12.2.4-2). Applying (12.2.3-6), Equation (12.2.4-4) is equivalently:

$$\delta\underline{g}_P^E = F(h)\,\frac{g}{R}\,\underline{u}_{ZN}^E\,\delta R - \frac{g}{R}\left(\delta\underline{R}^E - \underline{u}_{ZN}^E\,\delta R\right) + \delta\underline{g}_{Mdl}^E \tag{12.2.4-5}$$

The bracketed term in (12.2.4-5) should be recognized as:

$$\delta\underline{R}_H^E = \delta\underline{R}^E - \delta R\,\underline{u}_{ZN}^E \tag{12.2.4-6}$$

where

$\delta\underline{R}_H^E$ = Horizontal component (subscript designator) of $\delta\underline{R}^E$ in E Frame coordinates.

With (12.2.4-6), Equation (12.2.4-5) with F(h) from (12.2.4-2) becomes the final form:

$$\delta \underline{g}_P^E \approx -\frac{g}{R}\,\delta \underline{R}_H^E + F(h)\,\frac{g}{R}\,\underline{u}_{ZN}^E\,\delta R + \delta \underline{g}_{Mdl}^E$$
$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \tag{12.2.4-7}$$

A similar procedure applied to $\underline{g}_P^I$ in Equations (12.1.4-18) leads to:

$$\delta \underline{g}_P^I \approx -\frac{g}{R}\,\delta \underline{R}_H^I + F(h)\,\frac{g}{R}\,\underline{u}_{ZN}^I\,\delta R + \delta \underline{g}_{Mdl}^I$$
$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \tag{12.2.4-8}$$

where

$\delta \underline{g}_P^I$ = Error in $\underline{g}_P^I$.

$\delta \underline{g}_{Mdl}^I$ = Modeling error in $\underline{g}_P^I$ produced by variations in Equations (12.1-23) - (12.1-28) from the true earth model, variations in the Equations (12.1.4-18) gravity expression from Equations (12.1-23) - (12.1-28) and neglecting $\partial g_{P_{Up}}$ and $\partial g_{P_{North}}$ in Equations (12.1.4-18).

The error in the $\underline{\omega}_{EN}^N$ transport rate term is derived from the $\underline{\omega}_{EN}^N$ expression in Equations (12.1.2-6). From (12.1.2-6), the ∂G_C^N term in this expression is smaller by a factor of e than the other velocity term in this expression, hence, can be neglected for error analysis purposes. The differential of the result then is:

$$\delta \underline{\omega}_{EN}^N = \delta \rho_{ZN}\,\underline{u}_{ZN}^N + \frac{1}{r_l}\left(\underline{u}_{ZN}^N \times \delta \underline{v}^N\right) - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^N \times \underline{v}^N\right)\delta r_l \tag{12.2.4-9}$$

where

$\delta \underline{v}^N$ = As defined by Equation (12.2.2-2).

From Equations (12.1.2-6), r_l with the combined expressions for r_{ls}, R_S and $u_{ZN\,YE}$ is given by:

$$r_l = \left[1 + 2\left(2\,D_{23}^2 - 1\right)e\right]\left(1 - D_{23}^2\,e\right)R_0 + h \approx \left[1 + 2\left(\frac{3}{2}\,D_{23}^2 - 1\right)e\right]R_0 + h \tag{12.2.4-10}$$

The error in r_l is obtained as the differential of (12.2.4-10):

$$\delta r_l = 6\,e\,D_{23}\,\delta D_{23}\,R_0 + \delta h \tag{12.2.4-11}$$

The rationale leading to Equation (12.2.3-23) showed that $\delta D_{23}\,R_0$ is on the order of

$(R_0 / R)\ \delta \underline{R}^N$ which, from (12.2.3-23), is on the order of δh. Thus, Equation (12.2.4-11) simplifies to the approximate form:

$$\delta r_l \approx \delta h \tag{12.2.4-12}$$

With (12.2.4-12), Equation (12.2.4-9) becomes:

$$\delta \underline{\omega}_{EN}^N = \delta \rho_{ZN}\ \underline{u}_{ZN}^N + \frac{1}{r_l}\left(\underline{u}_{ZN}^N \times \delta \underline{v}_{ZN}^N\right) - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^N \times \underline{v}^N\right)\delta h \tag{12.2.4-13}$$

The $\delta \rho_{ZN}$ term in Equation (12.2.4-13) depends on the selection of ρ_{ZN} which, from Section 4.5, has several options. In this section, we will only address the "wander azimuth" and "free azimuth" approaches for selecting ρ_{ZN}.

From Section 4.5, for the "wander azimuth" implementation, $\rho_{ZN} = 0$, hence:

$$\delta \rho_{ZN} = 0 \qquad \text{For Wander Azimuth Implementation (See Section 4.5)} \tag{12.2.4-14}$$

For a "free azimuth" implementation, ρ_{ZN} is set equal to the negative of the vertical earth rate component. Analytically:

$$\rho_{ZN} = - \underline{u}_{ZN}^N \cdot \underline{\omega}_{IE}^N \tag{12.2.4-15}$$

or

$$\rho_{ZN} = - \underline{u}_{ZN}^N \cdot \left(C_E^N\ \underline{\omega}_{IE}^E\right) \qquad \text{For Free Azimuth Implementation (See Section 4.5)} \tag{12.2.4-16}$$

where

$\underline{\omega}_{IE}^N$ = Earth's rotation rate relative to inertial space (i.e., the angular rate of the E Frame relative to the I Frame) as projected onto N Frame axes.

$\underline{\omega}_{IE}^E$ = Earth's rotation rate relative to inertial space as projected onto E Frame axes.

Recognizing from Equations (12.1.2-6) that $\underline{u}_{ZN}^N$ and $\underline{\omega}_{IE}^E$ are constant, the differential of Equation (12.2.4-16) yields for $\delta \rho_{ZN}$:

$$\delta \rho_{ZN} = - \underline{u}_{ZN}^N \cdot \left(\delta C_E^N\ \underline{\omega}_{IE}^E\right) \tag{12.2.4-17}$$

or, with the transpose of Equation (12.2.1-11) for δC_E^N (and equating the transpose of $\left(\underline{\varepsilon}^N \times\right)$ to its negative):

$$\delta\rho_{ZN} = \underline{u}_{ZN}^N \cdot \left[\left(\underline{\varepsilon}^N \times\right) C_E^N \, \underline{\omega}_{IE}^E\right] \tag{12.2.4-18}$$

The desired expression for $\delta\rho_{ZN}$ in the free azimuth system is the equivalent compressed form of (12.2.4-18):

$$\delta\rho_{ZN} = -\left(\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \cdot \underline{u}_{ZN}^N \qquad \begin{array}{c}\text{For Free Azimuth Implementation}\\ \text{(See Section 4.5)}\end{array} \tag{12.2.4-19}$$

with

$$\underline{\omega}_{IE}^N = C_E^N \, \underline{\omega}_{IE}^E \tag{12.2.4-20}$$

and from Equations (12.1.2-6):

$$\underline{u}_{ZN}^N = \left[\, 0 \;\; 0 \;\; 1 \,\right]^T \qquad \underline{\omega}_{IE}^E = \left[\, 0 \;\; \omega_e \;\; 0 \,\right]^T \tag{12.2.4-21}$$

where

ω_e = Earth's rotation rate magnitude relative to inertial space.

The component form of (12.2.4-19) is provided by substituting (12.2.4-20) and (12.2.4-21) with the C_E^N components equal to the transpose of C_N^E as defined in Equations (4.4.1.1-2):

$$\delta\rho_{ZN} = \left(D_{22}\, \varepsilon_{XN} - D_{21}\, \varepsilon_{YN}\right) \omega_e \qquad \begin{array}{c}\text{For Free Azimuth Implementation}\\ \text{(See Section 4.5)}\end{array} \tag{12.2.4-22}$$

The vertical transport rate error component $\delta\rho_{ZN}$ in Equation (12.2.4-19) for the free azimuth system can be expressed alternatively as a function of $\delta\underline{R}^N$. The derivation is achieved by first defining $\underline{\omega}_{IE}^N$ and $\delta\underline{R}^N$ as the sum of their horizontal and vertical components:

$$\underline{\omega}_{IE}^N = \underline{\omega}_{IE_H}^N + \left(\underline{\omega}_{IE}^N \cdot \underline{u}_{ZN}^N\right) \underline{u}_{ZN}^N \tag{12.2.4-23}$$

$$\delta\underline{R}^N = \delta\underline{R}_H^N + \left(\delta\underline{R}^N \cdot \underline{u}_{ZN}^N\right) \underline{u}_{ZN}^N \tag{12.2.4-24}$$

where

H = Subscript designating horizontal component of the indicated N Frame vector which equals the vector with zero substituted for the vertical (Z) component.

Using equivalency Equation (12.2.3-19) for $\underline{\varepsilon}^N$ in terms of $\delta\underline{R}^N$, application of the generalized Equation (3.1.1-16) vector triple cross-product identity and substitution of (12.2.4-23) and (12.2.4-24) then yields for the $\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N$ term in (12.2.4-19):

$$
\begin{aligned}
\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N &= \left[\underline{\omega}_{IE_H}^N + \left(\underline{\omega}_{IE}^N \cdot \underline{u}_{ZN}^N\right)\underline{u}_{ZN}^N\right] \times \left\{\frac{1}{R}\,\underline{u}_{ZN}^N \times \left[\delta\underline{R}_H^N + \left(\delta\underline{R}^N \cdot \underline{u}_{ZN}^N\right)\underline{u}_{ZN}^N\right] + \varepsilon_{ZN}\,\underline{u}_{ZN}^N\right\} \\
&= \frac{1}{R}\left[\left(\underline{\omega}_{IE_H}^N \cdot \delta\underline{R}_H^N\right)\underline{u}_{ZN}^N - \left(\underline{\omega}_{IE}^N \cdot \underline{u}_{ZN}^N\right)\delta\underline{R}_H^N\right] + \varepsilon_{ZN}\,\underline{\omega}_{IE_H}^N \times \underline{u}_{ZN}^N
\end{aligned}
\qquad (12.2.4\text{-}25)
$$

In developing (12.2.4-25) use was made of the fact that the dot product of the vertical unit vector $\underline{u}_{ZN}^N$ with a horizontal vector is zero and the cross-product of $\underline{u}_{ZN}^N$ with itself is zero.

Substituting (12.2.4-25) into (12.2.4-19) and recognizing that $\underline{\omega}_{IE_H}^N \times \underline{u}_{ZN}^N$ is perpendicular to $\underline{u}_{ZN}^N$ (hence, has zero dot product with $\underline{u}_{ZN}^N$), then yields the desired expression for $\delta\rho_{ZN}$ for the free azimuth case in terms of $\delta\underline{R}^N$:

$$
\delta\rho_{ZN} = -\frac{1}{R}\,\underline{\omega}_{IE_H}^N \cdot \delta\underline{R}_H^N \qquad \text{For Free Azimuth Implementation (See Section 4.5)} \qquad (12.2.4\text{-}26)
$$

The $\underline{\omega}_{IE_H}^N$ term in (12.2.4-26) can be expressed from (12.2.4-20) as:

$$
\underline{\omega}_{IE_H}^N = \left(C_E^N\right)_H \underline{\omega}_{IE}^E \qquad (12.2.4\text{-}27)
$$

where

$\left(C_E^N\right)_H$ = Horizontal portion of C_E^N defined as C_E^N with row 3 (i.e., the vertical row) set to zero.

The component form of (12.2.4-26) is determined using (12.2.4-27) with (12.2.4-21) for $\underline{\omega}_{IE}^E$ and C_E^N as the transpose of (4.4.1.1-2) (i.e., the transpose of C_N^E):

$$
\delta\rho_{ZN} = -\frac{1}{R}\left(D_{21}\,\delta R_{XN} + D_{22}\,\delta R_{YN}\right)\omega_e \qquad \text{For Free Azimuth Implementation (See Section 4.5)} \qquad (12.2.4\text{-}28)
$$

where

$\delta R_{XN}, \delta R_{YN}$ = N Frame X, Y components of $\delta\underline{R}^N$ (and $\delta\underline{R}_H^N$).

12.2.5 BASIC NAVIGATION ERROR PARAMETER SELECTION

"Basic" navigation error parameters are defined herein as error parameters calculated by an integration process, in contrast with other error parameters that are calculated from the basic error parameters. The first step in inertial navigation system error analysis is the selection of a set of basic error parameters that best describe the error characteristics of concern, and which have no singularities in their integration process over the range of navigation conditions for which they will be utilized. The basic error parameters must also be such that their differential equations are completely defined in terms of the same basic parameters (and sensor error inputs). In Sections 12.2.1 - 12.2.3, several error parameters have been introduced for describing attitude/velocity/position navigation error. Classical error parameter groupings that lend themselves to use as basic attitude/velocity/position error parameters are ($\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh), ($\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$) and ($\underline{\psi}^N$, $\delta\underline{\upsilon}^N$, $\delta\underline{R}^N$); each provides a complete definition of the attitude/velocity/position error condition for which an independent set of differential equations can be derived (as is done in Section 12.3); each is free of singularities for any attitude/velocity/position condition. Other groupings are also possible, e.g., $\underline{\gamma}^N$, ε_{ZN}, $\delta\underline{V}^N$, $\delta\underline{R}^N$ discussed in Section 12.3.5. It is also possible to define the basic error parameters in coordinate frames other than the N Frame; e.g., Section 12.3.7.1 describes treatment of I Frame defined error parameters.

The selection of a particular set of basic error parameters depends on the unique requirements for each application. A principal consideration is whether the associated error parameter differential equations are to be applied analytically or by numerical integration techniques (e.g., in a computer as a simulation program or a Kalman filter application).

For analytical application, the choice of basic error parameters is typically based on whether their differential equations can be easily solved analytically for the error effects of interest (attitude, velocity or position) with minimum approximation. Occasionally, the requirement is that the differential equations themselves can be combined in a form that readily displays dynamic response characteristics without formal solution. Chapters 13 and 14 provide several examples of analytical exercises using various error parameter sets to describe particular strapdown inertial navigation system error performance characteristics under different conditions (e.g., general response characteristics in Sections 13.2, 13.2.1 and 13.2.2, long term approximate position error in Section 13.2.3, strapdown inertial sensor scale-factor/misalignment error effects in Section 13.2.4, navigation solutions for up to two hour cruise applications in Sections 13.3, 13.3.1 and 13.3.2, error effects during initial fine alignment in Chapter 14, etc.).

For numerical integration application, the choice of basic error parameters is typically based on whether the parameters (following determination by integration) can be converted into all equivalent forms required for the application (using the Section 12.2.1 - 12.2.3 conversion formulas), and the simplicity of the associated differential equations and required conversion equations (including approximations to be used, the number of basic error parameters, and required integration process execution rates to properly account for high frequency effects in the differential equation coefficients). An important consideration is whether or not integration/conversion software has been previously developed/validated for a particular error parameter set. It is also important to recognize from the onset that the navigation error parameters selected do not necessarily have to be direct representations of the errors in the particular strapdown inertial navigation attitude/velocity/position integration parameters for a given INS software configuration. Use of the Section 12.2.1 - 12.2.3 equivalency relationships allows conversion of the basic navigation error parameters to other forms when needed. For example, the $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh error parameters directly represent errors in C_B^N, $\underline{v}^N$, C_N^E, h used as the navigation integration parameters in some systems (See Equations (12.1.2-6)). If the navigation integration parameters are different (e.g., C_B^E, $\underline{v}^E$, $\underline{R}^E$ as in Equations (12.1.3-7)), the $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh errors can still be used as basic, with the Section 12.2.1 - 12.2.3 equivalency relationships allowing their conversion to other parameter forms (including the $\underline{\psi}^E$, $\delta\underline{V}^E$, $\delta\underline{R}^E$ set directly associated with C_B^E, $\underline{v}^E$, $\underline{R}^E$).

In many numerical integration applications, the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ set has been chosen as the basic navigation error parameters. The rationale has been that the minimum number of parameter components are utilized (9) to represent the attitude/velocity/position errors in three dimensions, the associated differential equations have no singularities, the differential equations are fairly simple compared to other error parameter sets (particularly for the $\underline{\psi}^N$ rate equation which is only a function of $\underline{\psi}^N$ and angular rate sensor error - See Section 12.3.3), and direct analytical equivalencies exist between $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ and other error parameters typically required in most applications. One exception to the last named condition is the case when the $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh parameters are required (representing errors defined in the N Frame) for which the $\underline{\varepsilon}^N$ vertical component (ε_{ZN}) is included (in addition to $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$). Thus, for complete generality, the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ parameters should include ε_{ZN} (the differential equation for which is provided in Equations (12.3.5-29) in terms of $\delta\underline{R}^N$). On the other hand, we can compute $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ and any other navigation error parameter from $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh. In selecting $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ or $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh as basic, it is important to note that ε_{ZN} is generally not a required output error parameter, and $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ can be converted to $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}_H^N$, δh (in which $\underline{\varepsilon}_H^N$ is the horizontal component of $\underline{\varepsilon}^N$), without ε_{ZN}.

Basic error parameter selection can be heavily influenced by particular numerical integration application requirements when computer throughput/memory limitations are at issue. For example, consider the case when attitude/velocity error is of paramount concern, position error is of minor importance and the application navigation period is fairly short (e.g., less than 1 hour). From Section 13.2.2 we learn that 84 minute dynamic Schuler oscillations are a characteristic part of the error behavior in all inertial navigation systems. In the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ parameter set, the Schuler dynamics are generated from $\delta\underline{R}^N$ coupling into the $\delta\underline{V}^N$ differential equation (see Equations (12.5.1-1)) while, for the $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh error set, Schuler dynamics are generated from $\delta\underline{v}^N$ coupling into the $\underline{\gamma}^N$ differential equation (see Equations (12.5.2-1)). To reduce throughput in this application, it is reasonable to neglect position error effects (based on the assumed requirements) and only consider the attitude/velocity parameters ($\underline{\psi}^N$, $\delta\underline{V}^N$ or $\underline{\gamma}^N$, $\delta\underline{v}^N$) as basic error parameter sets. For the $\underline{\gamma}^N$, $\delta\underline{v}^N$ set, Schuler dynamics would be modeled, but for the $\underline{\psi}^N$, $\delta\underline{V}^N$ set, Schuler dynamics would be absent (through elimination of $\delta\underline{R}^N$ to conserve throughput/memory). Thus, for this particular case, the $\underline{\gamma}^N$, $\delta\underline{v}^N$ error parameters appear preferable over the $\underline{\psi}^N$, $\delta\underline{V}^N$ set as better characterizing the Schuler dynamics.

As a general rule in numerical integration applications, Euler angle type error parameters (e.g., $\delta\phi$, $\delta\theta$, $\delta\psi_T$ roll/pitch/true heading error and δl, δL latitude/longitude error) and parameters defined in locally level east/north/up geographic coordinates (e.g., the $\delta\underline{\mathcal{V}}^{Geo}$ velocity error in Section 12.2.2) should be avoided in the basic error parameter set due to their inherent singularities.

12.3 NAVIGATION ERROR PARAMETER DIFFERENTIAL EQUATIONS

In this section we develop time rate differential equations for several groupings of the Section 12.2 navigation error parameters based on the Section 12.1 navigation equation sets. Section 12.3.1 describes general procedures for deriving the error parameter differential equations. Sections 12.3.2 - 12.3.7 then apply the procedures to derive various forms of the error parameter differential equations.

12.3.1 PROCEDURES FOR DEVELOPING ERROR PARAMETER DIFFERENTIAL EQUATIONS

In general, two methods can be used for developing differential equations for the navigation error parameters; a formal method and a direct linear differential method.

In the formal method two sets of navigation parameter differential equations are first defined; an idealized error free set, and a set of identical structure to the error free set which is implemented in the strapdown system computer. The strapdown computer implemented set is acknowledged to contain errors in the navigation parameters due to input and initialization errors (software and computer finite-word-length errors are not addressed in this section). The difference is then taken between the strapdown computer implemented differential equations and the equivalent idealized error free equations. The difference is identified as the error equations for the computer navigation parameters. In the differencing operation, error parameter products are dropped as second order, i.e., negligible. The resulting linearized differential error equation set is then reformatted to convert the derived navigation error parameters to the basic error parameters selected for error analysis (e.g., the error in a direction cosine matrix is typically converted to its equivalent rotation angle error vector equivalent such as in Equation (12.2.1-3)).

In the direct linear differential method, the analytical differential of the theoretical idealized error free navigation parameter differential equation is taken directly to obtain the linearized error parameter differential equations directly. As with the formal method, the differential error equations are then reformatted to convert the error parameters to the basic error parameters selected for error analysis.

Both the formal and direct linear differential methods generate the same end result, provided that the linearization process is carried out completely for the formal method. With the formal method, however, the option exists to retain second order terms for second order error analysis. While this is not generally required, there are those unusual cases when second order effects can be appreciable. In the subsections to follow we will use the direct differential method for simplicity. In this subsection, we provide an example of the formal method and its implications regarding the analysis of second order error effects. The example we choose is the formal method applied to the $\dot{C}_B^N$ expression in Equations (12.1.2-6):

$$\dot{C}_B^N = C_B^N \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IN}^N \times\right) C_B^N \tag{12.3.1-1}$$

The equivalent equation implemented in the strapdown navigation system computer is:

$$\dot{\hat{C}}_B^N = \hat{C}_B^N \left(\underline{\tilde{\omega}}_{IB}^B \times\right) - \left(\underline{\hat{\omega}}_{IN}^N \times\right) \hat{C}_B^N \tag{12.3.1-2}$$

where

$\tilde{}$ = Designation for actual parameter input to the system computer, hence, containing errors. The input parameter without the $\tilde{}$ designator is defined to be the idealized version of the input parameter, hence, error free.

$\widehat{\ }$ = Designation for parameter calculated in the system computer, hence, containing errors. The computed parameter without the $\widehat{\ }$ designator is defined to be the idealized version of the calculated parameter, hence, error free.

Equation (12.3.1-1) represents the idealized error free equation; Equation (12.3.1-2) with the $\sim$, $\widehat{\ }$ indicators is the system computer version containing errors. We define the relationship between the Equation (12.3.1-2) and (12.3.1-1) parameters as follows:

$$\hat{C}_B^N = C_B^N + \delta C_B^N \tag{12.3.1-3}$$

$$\underline{\tilde{\omega}}_{IB}^B = \underline{\omega}_{IB}^B + \delta\underline{\omega}_{IB}^B \tag{12.3.1-4}$$

$$\underline{\hat{\omega}}_{IN}^N = \underline{\omega}_{IN}^N + \delta\underline{\omega}_{IN}^N \tag{12.3.1-5}$$

where

$\delta(\ )$ = Error in the $\sim$ or $\widehat{\ }$ parameter. The $\delta(\ )$ definition provided here also applies for all subsections of Section 12.3.

Taking the difference between (12.3.1-2) and (12.3.1-1) and applying the (12.3.1-3) definition yields:

$$\delta\dot{C}_B^N = \hat{C}_B^N \left(\underline{\tilde{\omega}}_{IB}^B \times\right) - C_B^N \left(\underline{\omega}_{IB}^B \times\right) - \left[\left(\underline{\hat{\omega}}_{IN}^N \times\right) \hat{C}_B^N - \left(\underline{\omega}_{IN}^N \times\right) C_B^N\right] \tag{12.3.1-6}$$

Substituting (12.3.1-3) - (12.3.1-5) into (12.3.1-6) then gives:

$$\begin{aligned}
\delta\dot{C}_B^N &= \left(C_B^N + \delta C_B^N\right)\left[\left(\underline{\omega}_{IB}^B \times\right) + \left(\delta\underline{\omega}_{IB}^B \times\right)\right] - C_B^N \left(\underline{\omega}_{IB}^B \times\right) \\
&\qquad - \left\{\left[\left(\underline{\omega}_{IN}^N \times\right) + \left(\delta\underline{\omega}_{IN}^N \times\right)\right]\left(C_B^N + \delta C_B^N\right) - \left(\underline{\omega}_{IN}^N \times\right) C_B^N\right\} \\
&= \delta C_B^N \left(\underline{\omega}_{IB}^B \times\right) + C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right) C_B^N - \left(\underline{\omega}_{IN}^N \times\right) \delta C_B^N \\
&\qquad + \delta C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right) \delta C_B^N
\end{aligned} \tag{12.3.1-7}$$

Finally, the $\delta(\ )$ product terms in (12.3.1-7) are dropped as second order to obtain the linearized form:

$$\delta\dot{C}_B^N \approx \delta C_B^N \left(\underline{\omega}_{IB}^B \times\right) + C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right) C_B^N - \left(\underline{\omega}_{IN}^N \times\right) \delta C_B^N \tag{12.3.1-8}$$

Equation (12.3.1-8) is identical to the result that would have been obtained from the direct analytical differential of Equation (12.3.1-1). From Equation (12.3.1-8), appropriate substitutions are then made from Section 12.2 to generate the error differential equation in terms of desired basic error parameters.

Alternatively, Equations (12.3.1-6) can be expanded around the computer software navigation parameters by substitution of the converse of Equations (12.3.1-3) - (12.3.1-5) for the error free parameters:

$$\begin{aligned} \delta\dot{C}_B^N &= \hat{C}_B^N \left(\underline{\tilde{\omega}}_{IB}^B \times\right) - \left(\hat{C}_B^N - \delta C_B^N\right)\left[\left(\underline{\tilde{\omega}}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IB}^B \times\right)\right] \\ &\quad - \left\{\left(\underline{\hat{\omega}}_{IN}^N \times\right)\hat{C}_B^N - \left[\left(\underline{\hat{\omega}}_{IN}^N \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right)\right]\left(\hat{C}_B^N - \delta C_B^N\right)\right\} \\ &= \delta C_B^N \left(\underline{\tilde{\omega}}_{IB}^B \times\right) + \hat{C}_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right)\hat{C}_B^N - \left(\underline{\hat{\omega}}_{IN}^N \times\right)\delta C_B^N \\ &\quad - \delta C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) + \left(\delta\underline{\omega}_{IN}^N \times\right)\delta C_B^N \end{aligned} \tag{12.3.1-9}$$

When second order terms are dropped, Equation (12.3.1-9) becomes:

$$\delta\dot{C}_B^N = \delta C_B^N \left(\underline{\tilde{\omega}}_{IB}^B \times\right) + \hat{C}_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right)\hat{C}_B^N - \left(\underline{\hat{\omega}}_{IN}^N \times\right)\delta C_B^N \tag{12.3.1-10}$$

Equations (12.3.1-8) and (12.3.1-10) are equivalent to first order (i.e., if the $\tilde{}$ and $\hat{}$ parameters in (12.3.1-10) are replaced by their error free equivalents, which introduces only second order variations in the results). Note, however, that the second order terms in (12.3.1-9) that have been neglected in Equation (12.3.1-10) are the negative of the second order terms in (12.3.1-7) that were neglected in Equation (12.3.1-8). Thus, if second order error analysis is to be performed, it is important to accurately account for the linearized error equation form to be used (i.e., the Equation (12.3.1-10) form or the Equation (12.3.1-8) form). For simulation analyses, the (12.3.1-8) linearized form (with its associated (12.3.1-7) second order version) is usually appropriate because the idealized error free navigation parameters are usually available for usage. On the other hand, for implementation in a real-time system computer (e.g., the error propagation equations in a Kalman filter), only the computer version of the navigation parameters are usually available, hence, the Equation (12.3.1-10) form is the more appropriate version of the linearized error equations being utilized with its associated second order errors as characterized in Equation (12.3.1-9).

12.3.2 E FRAME DEFINED ERROR PARAMETER DIFFERENTIAL EQUATIONS

In this section, we derive a set of navigation error differential equations for the $\underline{\psi}$, $\delta\underline{V}$, $\delta\underline{R}$ attitude, velocity and position error parameters that were defined relative to the E Frame in Sections 12.2.1 - 12.2.3, and which represent the errors in the Equation (12.1.3-7) navigation parameters. In this section, the error equations will be developed in the E Frame. In the next section, we will develop their transformed equivalents in the N Frame, the frame more traditionally used for navigation error analysis.

We begin with the development of the $\underline{\psi}$ attitude error differential equation from the attitude expression in Equations (12.1.3-7):

$$\dot{C}_B^E = C_B^E \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IE}^E \times\right) C_B^E \tag{12.3.2-1}$$

Taking the analytical differential of (12.3.2-1), while recognizing from Equations (12.1.3-7) that the earth rate term in the E Frame is constant, yields:

$$\delta\dot{C}_B^E = \delta C_B^E \left(\underline{\omega}_{IB}^B \times\right) + C_B^E \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IE}^E \times\right) \delta C_B^E \tag{12.3.2-2}$$

in which the $\delta(\;)$ definition in Section 12.3.1 applies.

In order to reformat Equation (12.3.2-2) in terms of the $\underline{\psi}$ attitude error, we apply Equation (12.2.1-3) repeated below for δC_B^E:

$$\delta C_B^E = - \left(\underline{\psi}^E \times\right) C_B^E \tag{12.3.2-3}$$

The formal definition of the $\underline{\psi}$ attitude error is given by Equation (12.2.1-4).

The derivative of (12.3.2-3) is:

$$\delta\dot{C}_B^E = - \left(\dot{\underline{\psi}}^E \times\right) C_B^E - \left(\underline{\psi}^E \times\right) \dot{C}_B^E \tag{12.3.2-4}$$

or, with (12.3.2-1):

$$\begin{aligned} \delta\dot{C}_B^E &= - \left(\dot{\underline{\psi}}^E \times\right) C_B^E - \left(\underline{\psi}^E \times\right) \left[C_B^E \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IE}^E \times\right) C_B^E \right] \\ &= - \left(\dot{\underline{\psi}}^E \times\right) C_B^E - \left(\underline{\psi}^E \times\right) C_B^E \left(\underline{\omega}_{IB}^B \times\right) + \left(\underline{\psi}^E \times\right) \left(\underline{\omega}_{IE}^E \times\right) C_B^E \end{aligned} \tag{12.3.2-5}$$

Substituting (12.3.2-5) and (12.3.2-3) into (12.3.2-2) yields:

$$\begin{aligned} - \left(\dot{\underline{\psi}}^E \times\right) C_B^E - \left(\underline{\psi}^E \times\right) C_B^E \left(\underline{\omega}_{IB}^B \times\right) + \left(\underline{\psi}^E \times\right) \left(\underline{\omega}_{IE}^E \times\right) C_B^E & \\ = - \left(\underline{\psi}^E \times\right) C_B^E \left(\underline{\omega}_{IB}^B \times\right) + C_B^E \left(\delta\underline{\omega}_{IB}^B \times\right) + \left(\underline{\omega}_{IE}^E \times\right) \left(\underline{\psi}^E \times\right) C_B^E & \end{aligned} \tag{12.3.2-6}$$

or

$$- \left(\dot{\underline{\psi}}^E \times\right) C_B^E + \left(\underline{\psi}^E \times\right) \left(\underline{\omega}_{IE}^E \times\right) C_B^E = C_B^E \left(\delta\underline{\omega}_{IB}^B \times\right) + \left(\underline{\omega}_{IE}^E \times\right) \left(\underline{\psi}^E \times\right) C_B^E \tag{12.3.2-7}$$

Multiplying (12.3.2-7) on the right by the negative inverse of C_B^E and recognizing that the inverse of an idealized direction cosine matrix equals its transpose then obtains after rearrangement:

$$\left(\underline{\dot{\psi}}^E \times\right) = - C_B^E \left(\delta\underline{\omega}_{IB}^B \times\right)\left(C_B^E\right)^T + \left(\underline{\psi}^E \times\right)\left(\underline{\omega}_{IE}^E \times\right) - \left(\underline{\omega}_{IE}^E \times\right)\left(\underline{\psi}^E \times\right) \tag{12.3.2-8}$$

With (3.1.1-22) and (3.1.1-38), (12.3.2-8) becomes:

$$\left(\underline{\dot{\psi}}^E \times\right) = - \left[\left(C_B^E \, \delta\underline{\omega}_{IB}^B\right) \times\right] + \left[\left(\underline{\psi}^E \times \underline{\omega}_{IE}^E\right) \times\right] \tag{12.3.2-9}$$

or with (3.1.1-8) in the equivalent vector form:

$$\underline{\dot{\psi}}^E = - C_B^E \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IE}^E \times \underline{\psi}^E \tag{12.3.2-10}$$

Equation (12.3.2-10) is the E Frame error form of attitude rate Equation (12.3.2-1) in terms of the $\underline{\psi}$ error parameter. The $\delta\underline{\omega}_{IB}^B$ term in (12.3.2-10) represents the error in the strapdown angular rate sensor input data to the strapdown inertial navigation system.

We now direct our attention at the development of the $\delta\underline{V}$ velocity error differential equation from the Equations (12.1.3-7) velocity rate expression (and related inputs) repeated below:

$$\underline{\dot{v}}^E = C_B^E \, \underline{a}_{SF}^B + \underline{g}_P^E - 2 \, \underline{\omega}_{IE}^E \times \underline{v}^E - e_{vc_1} \, \underline{u}_{ZN}^E \tag{12.3.2-11}$$

$$\partial h = h - h_{Prsr} \tag{12.3.2-12}$$

$$e_{vc_1} = e_{vc_3} + C_2 \, \partial h \tag{12.3.2-13}$$

$$\dot{e}_{vc_3} = C_1 \, \partial h \tag{12.3.2-14}$$

The formal definition for the $\delta\underline{V}$ velocity error is given in the E Frame by Equation (12.2.2-1):

$$\delta\underline{V}^E \equiv \hat{\underline{v}}^E - \underline{v}^E \tag{12.3.2-15}$$

Taking the analytical differential of Equation (12.3.2-11) and identifying the differential of $\underline{v}^E$ as $\delta\underline{V}^E$ from (12.3.2-15) yields:

$$\delta\underline{\dot{V}}^E = C_B^E \, \delta\underline{a}_{SF}^B + \delta C_B^E \, \underline{a}_{SF}^B + \delta\underline{g}_P^E - 2 \, \underline{\omega}_{IE}^E \times \delta\underline{V}^E - \delta e_{vc_1} \, \underline{u}_{ZN}^E - e_{vc_1} \, \delta\underline{u}_{ZN}^E \tag{12.3.2-16}$$

The δC_B^E term in (12.3.2-16) is given by Equation (12.3.2-3). The $\delta \underline{g}_P^E$ term in (12.3.2-16) is given by Equations (12.2.4-7) with (12.2.3-14) and (12.2.4-6):

$$\delta \underline{g}_P^E \approx -\frac{g}{R}\,\delta \underline{R}_H^E + F(h)\,\frac{g}{R}\,\underline{u}_{ZN}^E\,\delta R + \delta \underline{g}_{Mdl}^E$$
$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \tag{12.3.2-17}$$

with

$$\delta R = \underline{u}_{ZN}^E \cdot \delta \underline{R}^E \tag{12.3.2-18}$$

$$\delta \underline{R}_H^E = \delta \underline{R}^E - \delta R\,\underline{u}_{ZN}^E \tag{12.3.2-19}$$

The $\delta \underline{u}_{ZN}^E$ term in (12.3.2-16) is given by Equation (12.2.3-11) combined with $\left(\underline{\varepsilon}^N \times \underline{u}_{ZN}^N\right)$ from the rearranged form of (12.2.3-16) in the E Frame. Including substitution from (12.3.2-19), this finds:

$$\delta \underline{u}_{ZN}^E = \frac{1}{R}\left(\delta \underline{R}^E - \delta R\,\underline{u}_{ZN}^E\right) = \frac{1}{R}\,\delta \underline{R}_H^E \tag{12.3.2-20}$$

Substituting (12.3.2-17), (12.3.2-20) and (12.3.2-3) into (12.3.2-16) then obtains:

$$\delta \dot{\underline{V}}^E = C_B^E\,\delta \underline{a}_{SF}^B - \left(\underline{\psi}^E \times\right) C_B^E\,\underline{a}_{SF}^B - \frac{g}{R}\,\delta \underline{R}_H^E + F(h)\,\frac{g}{R}\,\underline{u}_{ZN}^E\,\delta R + \delta \underline{g}_{Mdl}^E$$
$$- 2\,\underline{\omega}_{IE}^E \times \delta \underline{V}^E - \delta e_{vc_1}\,\underline{u}_{ZN}^E - e_{vc_1}\,\frac{1}{R}\,\delta \underline{R}_H^E \tag{12.3.2-21}$$
$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

The $e_{vc_1}\,\frac{1}{R}\,\delta \underline{R}_H^E$ term in (12.3.2-21) can be shown to be negligible by noting an important characteristic of the Equation (12.3.2-12) - (12.3.2-14) ∂h vertical loop control signal; namely that ∂h is a direct measurement of first order altitude error effects. This is easily demonstrated by recognizing that ∂h is actually the difference between the system calculated altitude and the input pressure altitude (both containing errors):

$$\partial h = \widehat{h} - \widetilde{h}_{Prsr} \tag{12.3.2-22}$$

However, $\widehat{h}$ and $\widetilde{h}_{Prsr}$ can be defined as equal to the true altitude h plus variations (errors) in $\widehat{h}$ and $\widetilde{h}_{Prsr}$ from the true altitude:

$$\widehat{h} = h_{True} + \delta h \qquad\qquad \widetilde{h}_{Prsr} = h_{True} + \delta h_{Prsr} \tag{12.3.2-23}$$

where

h_{True} = Correct (error free) altitude.

δh_{Prsr} = Error in the input pressure altitude signal.

Thus, with (12.3.2-23), Equation (12.3.2-22) becomes:

$$\partial h = \delta h - \delta h_{Prsr} \tag{12.3.2-24}$$

On the other hand, from the differential of Equation (12.3.2-12), we also see that:

$$\delta(\partial h) = \delta h - \delta h_{Prsr} \tag{12.3.2-25}$$

Therefore, from (12.3.2-24) and (12.3.2-25):

$$\delta(\partial h) = \partial h \tag{12.3.2-26}$$

Substituting (12.3.2-26) into the differential of (12.3.2-14) shows that:

$$\delta\dot{e}_{vc_3} = C_1\,\delta(\partial h) = C_1\,\partial h = \dot{e}_{vc_3} \tag{12.3.2-27}$$

or

$$\delta e_{vc_3} = e_{vc_3} \tag{12.3.2-28}$$

Substituting (12.3.2-26), (12.3.2-28) and (12.3.2-13) into the differential of (12.3.2-13) then gives:

$$\delta e_{vc_1} = \delta e_{vc_3} + C_2\,\delta(\partial h) = e_{vc_3} + C_2\,\partial h = e_{vc_3} + e_{vc_1} - e_{vc_3} \tag{12.3.2-29}$$

or

$$\delta e_{vc_1} = e_{vc_1} \tag{12.3.2-30}$$

With Equation (12.3.2-30) in (12.3.2-21), it should be clear that the $e_{vc_1}\,\frac{1}{R}\,\delta\underline{R}_H^E$ term is small compared to the $\delta e_{vc_1}\,\underline{u}_{ZN}^E$ term. This forms the basis for dropping the $e_{vc_1}\,\frac{1}{R}\,\delta\underline{R}_H^E$ term in Equation (12.3.2-21) as negligible.

Finally, the differential of Equation (12.3.2-13) with (12.3.2-25) shows that:

$$\delta e_{vc_1} = \delta e_{vc_3} + C_2\,\delta(\partial h) = \delta e_{vc_3} + C_2\left(\delta h - \delta h_{Prsr}\right) \tag{12.3.2-31}$$

The δh altitude error term in (12.3.2-31) can be expressed in terms of $\delta \underline{R}$ error parameters using the Equation (12.2.3-23) equivalency between altitude error and the vertical component of $\delta \underline{R}$:

$$\delta h \; = \; \delta R \tag{12.3.2-32}$$

Thus, in terms of the $\delta \underline{R}$ error parameter, δe_{vc_1} from Equation (12.3.2-31) becomes:

$$\delta e_{vc_1} \; = \; \delta e_{vc_3} + C_2 \left(\delta R - \delta h_{Prsr} \right) \tag{12.3.2-33}$$

We now substitute (12.3.2-33) into (12.3.2-21), drop the $e_{vc_1} \frac{1}{R} \delta \underline{R}_H^E$ term as negligible, collect and condense terms to obtain the final form of the $\delta \underline{V}^E$ velocity error rate equation:

$$\begin{aligned} \delta \dot{\underline{V}}^E \; = \; & C_B^E \, \delta \underline{a}_{SF}^B + \underline{a}_{SF}^E \times \underline{\psi}^E - \frac{g}{R} \, \delta \underline{R}_H^E - 2 \, \underline{\omega}_{IE}^E \times \delta \underline{V}^E + \delta \underline{g}_{Mdl}^E \\ & + \left[\left(F(h) \, \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^E \end{aligned} \tag{12.3.2-34}$$

$$F(h) \; = \; 2 \quad \text{For} \quad h \; \geq \; 0 \qquad\qquad F(h) \; = \; -1 \quad \text{For} \quad h \; < \; 0$$

The $\delta \underline{a}_{SF}^B$ term in (12.3.2-34) represents the error in the strapdown accelerometer signals input to the strapdown inertial navigation system. The δe_{vc_3} term for (12.3.2-34) is obtained from the differential of Equations (12.3.2-14) with (12.3.2-25) and (12.3.2-32):

$$\delta \dot{e}_{vc_3} \; = \; C_1 \left(\delta R - \delta h_{Prsr} \right) \tag{12.3.2-35}$$

It remains to determine the $\delta \underline{R}^E$ position error differential equation. We begin with the E Frame position rate expression and its contributor in Equations (12.1.3-7):

$$\dot{\underline{R}}^E \; = \; \underline{v}^E - e_{vc_2} \, \underline{u}_{ZN}^E \tag{12.3.2-36}$$

$$e_{vc_2} \; = \; C_3 \, \partial h \tag{12.3.2-37}$$

Taking the differential of (12.3.2-36) yields:

$$\delta \dot{\underline{R}}^E \; = \; \delta \underline{V}^E - \delta e_{vc_2} \, \underline{u}_{ZN}^E - e_{vc_2} \, \delta \underline{u}_{ZN}^E \tag{12.3.2-38}$$

or, with (12.3.2-20):

$$\delta \dot{\underline{R}}^E \; = \; \delta \underline{V}^E - \delta e_{vc_2} \, \underline{u}_{ZN}^E - e_{vc_2} \, \frac{1}{R} \, \delta \underline{R}_H^E \tag{12.3.2-39}$$

Following the same rationale that led to Equation (12.3.2-28), we can show that:

$$\delta e_{vc_2} = e_{vc_2} \tag{12.3.2-40}$$

From (12.3.2-40), we see that the $e_{vc_2} \frac{1}{R} \delta \underline{R}_H^E$ term in (12.3.2-39) is small compared to the $\delta e_{vc_2} \underline{u}_{ZN}^E$ term, hence, can be neglected. We also note from the differential of (12.3.2-37) with (12.3.2-25) and (12.3.2-32) that:

$$\delta e_{vc_2} = C_3 \left(\delta R - \delta h_{Prsr} \right) \tag{12.3.2-41}$$

Lastly, substituting (12.3.2-41) into (12.3.2-39) and neglecting $e_{vc_2} \frac{1}{R} \delta \underline{R}_H^E$, we obtain the $\delta \underline{R}^E$ position error rate equation:

$$\delta \dot{\underline{R}}^E = \delta \underline{V}^E - C_3 \left(\delta R - \delta h_{Prsr} \right) \underline{u}_{ZN}^E \tag{12.3.2-42}$$

In summary, the attitude, velocity and position error rate equations that characterize the errors in navigation Equations (12.1.3-7) are given by Equations (12.3.2-10), (12.3.2-34), (12.3.2-35) and (12.3.2-42) with (12.3.2-18) and (12.3.2-19). These equations are repeated below for easy reference.

$$\dot{\underline{\psi}}^E = - C_B^E \, \delta \underline{\omega}_{IB}^B - \underline{\omega}_{IE}^E \times \underline{\psi}^E$$

$$\delta \dot{\underline{V}}^E = C_B^E \, \delta \underline{a}_{SF}^B + \underline{a}_{SF}^E \times \underline{\psi}^E - \frac{g}{R} \, \delta \underline{R}_H^E - 2 \, \underline{\omega}_{IE}^E \times \delta \underline{V}^E + \delta \underline{g}_{Mdl}^E$$
$$+ \left[\left(F(h) \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^E$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \tag{12.3.2-43}$$

$$\delta \dot{\underline{R}}^E = \delta \underline{V}^E - C_3 \left(\delta R - \delta h_{Prsr} \right) \underline{u}_{ZN}^E$$

$$\delta \dot{e}_{vc_3} = C_1 \left(\delta R - \delta h_{Prsr} \right)$$

$$\delta \underline{R}_H^E = \delta \underline{R}^E - \delta R \, \underline{u}_{ZN}^E$$

$$\delta R = \underline{u}_{ZN}^E \cdot \delta \underline{R}^E$$

12.3.3 E FRAME DEFINED ERROR PARAMETER DIFFERENTIAL EQUATIONS TRANSFORMED TO THE N FRAME

Error Equations (12.3.2-43) had their error parameters ($\underline{\psi}$, $\delta\underline{V}$ and $\delta\underline{R}$) defined in the E Frame (by Equations (12.2.1-4), (12.2.2-1) and (12.2.3-1)) and their differential rate equations written in the E Frame. The integral of Equations (12.3.2-43) generates $\underline{\psi}$, $\delta\underline{V}$, $\delta\underline{R}$ in E Frame coordinates. We can also transform Equations (12.3.2-43) to the N Frame to generate an equivalent set of differential equations that when integrated in the N Frame, yield $\underline{\psi}$, $\delta\underline{V}$, $\delta\underline{R}$ as projected on N Frame axes.

We begin with the $\dot{\underline{\psi}}^E$ expression from Equations (12.3.2-43):

$$\dot{\underline{\psi}}^E = - C_B^E \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IE}^E \times \underline{\psi}^E \tag{12.3.3-1}$$

Applying generalized Equation (3.4-6) to (12.3.3-1) obtains the transformed equivalent in the N Frame:

$$\begin{aligned}
\dot{\underline{\psi}}^N &= C_E^N \, \dot{\underline{\psi}}^E + \underline{\omega}_{NE}^N \times \underline{\psi}^N \\
&= C_E^N \left(- C_B^E \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IE}^E \times \underline{\psi}^E \right) + \underline{\omega}_{NE}^N \times \underline{\psi}^N \\
&= - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IE}^N \times \underline{\psi}^N + \underline{\omega}_{NE}^N \times \underline{\psi}^N \\
&= - C_B^N \, \delta\underline{\omega}_{IB}^B - \left(\underline{\omega}_{IE}^N - \underline{\omega}_{NE}^N \right) \times \underline{\psi}^N
\end{aligned} \tag{12.3.3-2}$$

where

$\underline{\omega}_{NE}^N$ = Angular rate of Frame E relative to Frame N (subscript designation) as projected on N Frame axes (superscript designation).

But we also know that:

$$\underline{\omega}_{NE}^N = - \underline{\omega}_{EN}^N \tag{12.3.3-3}$$

where

$\underline{\omega}_{EN}^N$ = Angular rate of Frame N relative to Frame E as projected on N Frame axes.

From (12.3.3-3), the bracketed angular rate term in (12.3.3-2) becomes:

$$\underline{\omega}_{IE}^N - \underline{\omega}_{NE}^N = \underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N = \underline{\omega}_{IN}^N \tag{12.3.3-4}$$

where

$\underline{\omega}_{IE}^{N}$ = Angular rate of Frame E relative to inertial Frame I (i.e., earth rate) as projected on N Frame axes.

$\underline{\omega}_{IN}^{N}$ = Angular rate of Frame N relative to Frame I as projected on N Frame axes.

With (12.3.3-4), Equation (12.3.3-2) simplifies to the final form:

$$\dot{\underline{\psi}}^{N} = - C_B^N \, \delta\underline{\omega}_{IB}^{B} - \underline{\omega}_{IN}^{N} \times \underline{\psi}^{N} \tag{12.3.3-5}$$

The same procedure can also be applied to the $\delta\dot{\underline{V}}^{E}$ and $\delta\dot{\underline{R}}^{E}$ expressions in Equations (12.3.2-43) and their supporting expressions. The final result including Equation (12.3.3-5) is summarized below:

$$\begin{aligned}
&\dot{\underline{\psi}}^{N} = - C_B^N \, \delta\underline{\omega}_{IB}^{B} - \underline{\omega}_{IN}^{N} \times \underline{\psi}^{N} \\
&\delta\dot{\underline{V}}^{N} = C_B^N \, \delta\underline{a}_{SF}^{B} + \underline{a}_{SF}^{N} \times \underline{\psi}^{N} - \frac{g}{R} \, \delta\underline{R}_{H}^{N} - \left(\underline{\omega}_{IE}^{N} + \underline{\omega}_{IN}^{N}\right) \times \delta\underline{V}^{N} + \delta\underline{g}_{Mdl}^{N} \\
&\qquad + \left[\left(F(h) \, \frac{g}{R} - C_2\right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^{N} \\
&F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \\
&\delta\dot{\underline{R}}^{N} = \delta\underline{V}^{N} - \underline{\omega}_{EN}^{N} \times \delta\underline{R}^{N} - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^{N} \\
&\delta\dot{e}_{vc_3} = C_1 \left(\delta R - \delta h_{Prsr}\right) \\
&\delta\underline{R}_{H}^{N} = \delta\underline{R}^{N} - \delta R \, \underline{u}_{ZN}^{N} \\
&\delta R = \underline{u}_{ZN}^{N} \cdot \delta\underline{R}^{N}
\end{aligned} \tag{12.3.3-6}$$

Equations (12.3.3-6) represent the error form of navigation Equations (12.1.3-7) as projected on N Frame axes.

12.3.4 N FRAME DEFINED ERROR PARAMETER DIFFERENTIAL EQUATIONS

In this section we develop the N Frame differential equations for the errors in navigation Equations (12.1.2-6) as defined in the N Frame by attitude, velocity, position error parameters $\underline{\gamma}$, $\delta\underline{v}$, $\underline{\varepsilon}$, δh. We begin with the attitude rate expression and its supporting elements from Equations (12.1.2-6):

$$\dot{C}_B^N = C_B^N \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IN}^N \times\right) C_B^N \tag{12.3.4-1}$$

$$\underline{\omega}_{IN}^N = \underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N \tag{12.3.4-2}$$

$$\underline{\omega}_{IE}^N = C_E^N \, \underline{\omega}_{IE}^E \tag{12.3.4-3}$$

Taking the differential of (12.3.4-1);

$$\delta\dot{C}_B^N = \delta C_B^N \left(\underline{\omega}_{IB}^B \times\right) + C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right) C_B^N - \left(\underline{\omega}_{IN}^N \times\right) \delta C_B^N \tag{12.3.4-4}$$

The δC_B^N term in (12.3.4-4) is converted to its equivalent $\underline{\gamma}$ rotation angle error vector form using Equation (12.2.1-9) repeated below:

$$\delta C_B^N = - \left(\underline{\gamma}^N \times\right) C_B^N \tag{12.3.4-5}$$

The $\underline{\gamma}$ rotation angle error vector is defined formally by Equation (12.2.1-10). Taking the derivative of (12.3.4-5) and substituting $\dot{C}_B^N$ from (12.3.4-1) gives for $\delta\dot{C}_B^N$:

$$\begin{aligned} \delta\dot{C}_B^N &= - \left(\dot{\underline{\gamma}}^N \times\right) C_B^N - \left(\underline{\gamma}^N \times\right) \dot{C}_B^N \\ &= - \left(\dot{\underline{\gamma}}^N \times\right) C_B^N - \left(\underline{\gamma}^N \times\right) C_B^N \left(\underline{\omega}_{IB}^B \times\right) + \left(\underline{\gamma}^N \times\right) \left(\underline{\omega}_{IN}^N \times\right) C_B^N \end{aligned} \tag{12.3.4-6}$$

We now substitute (12.3.4-6) for $\delta\dot{C}_B^N$ and (12.3.4-5) for δC_B^N in Equation (12.3.4-4):

$$\begin{aligned} &- \left(\dot{\underline{\gamma}}^N \times\right) C_B^N - \left(\underline{\gamma}^N \times\right) C_B^N \left(\underline{\omega}_{IB}^B \times\right) + \left(\underline{\gamma}^N \times\right) \left(\underline{\omega}_{IN}^N \times\right) C_B^N \\ &\quad = - \left(\underline{\gamma}^N \times\right) C_B^N \left(\underline{\omega}_{IB}^B \times\right) + C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) - \left(\delta\underline{\omega}_{IN}^N \times\right) C_B^N + \left(\underline{\omega}_{IN}^N \times\right) \left(\underline{\gamma}^N \times\right) C_B^N \end{aligned} \tag{12.3.4-7}$$

or upon cancellation of like terms and rearrangement:

$$\begin{aligned} \left(\dot{\underline{\gamma}}^N \times\right) C_B^N &= - C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right) + \left(\underline{\gamma}^N \times\right) \left(\underline{\omega}_{IN}^N \times\right) C_B^N \\ &\quad - \left(\underline{\omega}_{IN}^N \times\right) \left(\underline{\gamma}^N \times\right) C_B^N + \left(\delta\underline{\omega}_{IN}^N \times\right) C_B^N \end{aligned} \tag{12.3.4-8}$$

Multiplying (12.3.4-8) on the right by the inverse of C_B^N, recognizing that the inverse of an idealized direction cosine matrix equals its transpose, and application of generalized Equations (3.1.1-22) and (3.1.1-38) then yields:

$$\begin{aligned}\left(\dot{\underline{\gamma}}^N \times\right) &= - C_B^N \left(\delta\underline{\omega}_{IB}^B \times\right)\left(C_B^N\right)^T + \left(\underline{\gamma}^N \times\right)\left(\underline{\omega}_{IN}^N \times\right) - \left(\underline{\omega}_{IN}^N \times\right)\left(\underline{\gamma}^N \times\right) + \left(\delta\underline{\omega}_{IN}^N \times\right) \\ &= - \left[\left(C_B^N \,\delta\underline{\omega}_{IB}^B\right)\times\right] + \left[\left(\underline{\gamma}^N \times \underline{\omega}_{IN}^N\right)\times\right] + \left(\delta\underline{\omega}_{IN}^N \times\right)\end{aligned} \tag{12.3.4-9}$$

or, with (3.1.1-8), in the equivalent vector form:

$$\dot{\underline{\gamma}}^N = - C_B^N \,\delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \delta\underline{\omega}_{IN}^N \tag{12.3.4-10}$$

The $\delta\underline{\omega}_{IN}^N$ term in (12.3.4-10) is determined from the differential of Equations (12.3.4-2) and (12.3.4-3). Recognizing from Equations (12.1.2-6) that $\underline{\omega}_{IE}^E$ is constant yields:

$$\delta\underline{\omega}_{IN}^N = \delta\underline{\omega}_{IE}^N + \delta\underline{\omega}_{EN}^N \tag{12.3.4-11}$$

$$\delta\underline{\omega}_{IE}^N = \delta C_E^N \,\underline{\omega}_{IE}^E \tag{12.3.4-12}$$

Substituting the transpose of δC_N^E from (12.2.1-11) for δC_E^N in (12.3.4-12), and equating the transpose of the skew symmetric matrix $\left(\underline{\varepsilon}^N \times\right)$ to its negative obtains:

$$\delta\underline{\omega}_{IE}^N = - \left(\underline{\varepsilon}^N \times\right) C_E^N \,\underline{\omega}_{IE}^E = - \left(\underline{\varepsilon}^N \times\right) \underline{\omega}_{IE}^N = \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N \tag{12.3.4-13}$$

With (12.3.4-11) and (12.3.4-13), Equation (12.3.4-10) assumes the final form:

$$\dot{\underline{\gamma}}^N = - C_B^N \,\delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \tag{12.3.4-14}$$

The $\delta\underline{\omega}_{EN}^N$ term in (12.3.4-14) is provided by Equation (12.2.4-13) with (12.2.4-14) and (12.2.4-19) repeated below:

$$\delta\underline{\omega}_{EN}^N = \delta\rho_{ZN} \,\underline{u}_{ZN}^N + \frac{1}{r_l}\left(\underline{u}_{ZN}^N \times \delta\underline{v}^N\right) - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^N \times \underline{v}^N\right)\delta h \tag{12.3.4-15}$$

$$\delta\rho_{ZN} = 0 \qquad \text{For Wander Azimuth Implementation} \tag{12.3.4-16}$$

$$\delta\rho_{ZN} = - \left(\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \cdot \underline{u}_{ZN}^N \qquad \text{For Free Azimuth Implementation} \tag{12.3.4-17}$$

Development of the N Frame defined $\delta\underline{v}$ velocity error rate equation begins with the velocity expression and its supporting elements from Equations (12.1.2-6):

$$\dot{\underline{v}}^N = C_B^N \, \underline{a}_{SF}^B + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N\right) \times \underline{v}^N - e_{vc_1} \, \underline{u}_{ZN}^N \qquad (12.3.4\text{-}18)$$

$$\partial h = h - h_{Prsr} \qquad (12.3.4\text{-}19)$$

$$e_{vc_1} = e_{vc_3} + C_2 \, \partial h = e_{vc_3} + C_2 \left(h - h_{Prsr}\right) \qquad (12.3.4\text{-}20)$$

$$\dot{e}_{vc_3} = C_1 \, \partial h = C_1 \left(h - h_{Prsr}\right) \qquad (12.3.4\text{-}21)$$

Taking the differential of (12.3.4-18) while recognizing from (12.1.2-6) that $\underline{u}_{ZN}^N$ is constant, gives:

$$\begin{aligned} \delta\dot{\underline{v}}^N = {} & C_B^N \, \delta\underline{a}_{SF}^B + \delta C_B^N \, \underline{a}_{SF}^B + \delta\underline{g}_P^N - \left(\delta\underline{\omega}_{EN}^N + 2\, \delta\underline{\omega}_{IE}^N\right) \times \underline{v}^N \\ & - \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N\right) \times \delta\underline{v}^N - \delta e_{vc_1} \, \underline{u}_{ZN}^N \end{aligned} \qquad (12.3.4\text{-}22)$$

The δe_{vc_1} term in (12.3.4-22) is obtained from the differential of (12.3.4-20):

$$\delta e_{vc_1} = \delta e_{vc_3} + C_2 \left(\delta h - \delta h_{Prsr}\right) \qquad (12.3.4\text{-}23)$$

The $\delta\underline{g}_P^N$ term in (12.3.4-22) is from Equation (12.2.4-2):

$$\begin{gathered} \delta\underline{g}_P^N \approx F(h) \, \frac{g}{R} \, \underline{u}_{ZN}^N \, \delta h + \delta\underline{g}_{Mdl}^N \\ F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \end{gathered} \qquad (12.3.4\text{-}24)$$

Substituting (12.3.4-23), (12.3.4-24), δC_B^N from (12.3.4-5) and $\delta\underline{\omega}_{IE}^N$ from (12.3.4-13) into (12.3.4-22) then yields:

$$\begin{aligned} \delta\dot{\underline{v}}^N = {} & C_B^N \, \delta\underline{a}_{SF}^B - \left(\underline{\gamma}^N \times\right) C_B^N \, \underline{a}_{SF}^B + F(h) \, \frac{g}{R} \, \underline{u}_{ZN}^N \, \delta h + \delta\underline{g}_{Mdl}^N \\ & - \left(\delta\underline{\omega}_{EN}^N - 2 \left(\underline{\varepsilon}^N \times\right) \underline{\omega}_{IE}^N\right) \times \underline{v}^N - \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N\right) \times \delta\underline{v}^N \\ & - \left[\delta e_{vc_3} + C_2 \left(\delta h - \delta h_{Prsr}\right)\right] \underline{u}_{ZN}^N \\ F(h) = {} & 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \end{aligned} \qquad (12.3.4\text{-}25)$$

or after compression and rearrangement, the final form:

$$\begin{aligned}\delta\dot{\underline{v}}^N &= C_B^N\,\delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N + \underline{v}^N \times \left(\delta\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \\ &\quad - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \delta\underline{v}^N + \delta\underline{g}_{Mdl}^N \\ &\quad + \left[\left(F(h)\,\frac{g}{R} - C_2\right)\delta h + C_2\,\delta h_{Prsr} - \delta e_{vc_3}\right]\underline{u}_{ZN}^N\end{aligned} \tag{12.3.4-26}$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

The $\delta\underline{\omega}_{EN}^N$ term in (12.3.4-26) is provided by Equations (12.3.4-15) - (12.3.4-17). The δe_{vc_3} term in (12.3.4-26) is provided by the differential of Equation (12.3.4-21):

$$\delta\dot{e}_{vc_3} = C_1\left(\delta h - \delta h_{Prsr}\right) \tag{12.3.4-27}$$

Development of the $\underline{\varepsilon}$, δh position error terms begins with a restatement of the position matrix and altitude rate terms with their supporting elements from Equations (12.1.2-6):

$$\dot{C}_N^E = C_N^E\left(\underline{\omega}_{EN}^N \times\right) \tag{12.3.4-28}$$

$$\dot{h} = \underline{v}^N \cdot \underline{u}_{ZN}^N - e_{vc_2} \tag{12.3.4-29}$$

$$e_{vc_2} = C_3\,\partial h = C_3\left(h - h_{Prsr}\right) \tag{12.3.4-30}$$

Taking the differential of (12.3.4-28):

$$\delta\dot{C}_N^E = \delta C_N^E\left(\underline{\omega}_{EN}^N \times\right) + C_N^E\left(\delta\underline{\omega}_{EN}^N \times\right) \tag{12.3.4-31}$$

The δC_N^E term in (12.3.4-31) is from Equation (12.2.1-11):

$$\delta C_N^E = C_N^E\left(\underline{\varepsilon}^N \times\right) \tag{12.3.4-32}$$

The $\delta\dot{C}_N^E$ term in (12.3.4-31) is obtained from the time derivative of (12.3.4-32) with Equation (12.3.4-28) for $\dot{C}_N^E$:

$$\delta\dot{C}_N^E = \dot{C}_N^E\left(\underline{\varepsilon}^N \times\right) + C_N^E\left(\dot{\underline{\varepsilon}}^N \times\right) = C_N^E\left(\underline{\omega}_{EN}^N \times\right)\left(\underline{\varepsilon}^N \times\right) + C_N^E\left(\dot{\underline{\varepsilon}}^N \times\right) \tag{12.3.4-33}$$

Substituting (12.3.4-32) and (12.3.4-33) into Equation (12.3.4-31) then yields:

$$C_N^E\left(\underline{\omega}_{EN}^N \times\right)\left(\underline{\varepsilon}^N \times\right) + C_N^E\left(\dot{\underline{\varepsilon}}^N \times\right) = C_N^E\left(\underline{\varepsilon}^N \times\right)\left(\underline{\omega}_{EN}^N \times\right) + C_N^E\left(\delta\underline{\omega}_{EN}^N \times\right) \tag{12.3.4-34}$$

or upon rearrangement:

$$C_N^E \left(\dot{\underline{\varepsilon}}^N \times\right) = C_N^E \left(\underline{\varepsilon}^N \times\right) \left(\underline{\omega}_{EN}^N \times\right) - C_N^E \left(\underline{\omega}_{EN}^N \times\right)\left(\underline{\varepsilon}^N \times\right) + C_N^E \left(\delta\underline{\omega}_{EN}^N \times\right) \tag{12.3.4-35}$$

Multiplying (12.3.4-35) on the left by the inverse of C_N^E and applying generalized Equation (3.1.1-22) gives:

$$\begin{aligned} \left(\dot{\underline{\varepsilon}}^N \times\right) &= \left(\underline{\varepsilon}^N \times\right) \left(\underline{\omega}_{EN}^N \times\right) - \left(\underline{\omega}_{EN}^N \times\right)\left(\underline{\varepsilon}^N \times\right) + \left(\delta\underline{\omega}_{EN}^N \times\right) \\ &= \left[\left(\underline{\varepsilon}^N \times \underline{\omega}_{EN}^N\right) \times\right] + \left(\delta\underline{\omega}_{EN}^N \times\right) \end{aligned} \tag{12.3.4-36}$$

Equation (12.3.4-36) in vector form is the final result for the rate of change of $\underline{\varepsilon}^N$:

$$\dot{\underline{\varepsilon}}^N = - \underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \tag{12.3.4-37}$$

The $\delta\underline{\omega}_{EN}^N$ term in (12.3.4-37) is provided by Equations (12.3.4-15) - (12.3.4-17).

The δh altitude error rate equation is obtained from the differential of (12.3.4-29) with (12.3.4-30) while recognizing from Equations (12.1.2-6) that $\underline{u}_{ZN}^N$ is constant:

$$\delta\dot{h} = \underline{u}_{ZN}^N \cdot \delta\underline{v}^N - C_3 \left(\delta h - \delta h_{Prsr}\right) \tag{12.3.4-38}$$

In summary, the attitude, velocity and position error rate equations that characterize the errors in navigation Equations (12.1.2-6) are given by Equations (12.3.4-14), (12.3.4-26), (12.3.4-27), (12.3.4-37) and (12.3.4-38) with (12.3.4-15) - (12.3.4-17). These equations are repeated below for easy reference.

$$\begin{aligned} \dot{\underline{\gamma}}^N &= - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \\ \delta\dot{\underline{v}}^N &= C_B^N \, \delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N + \underline{v}^N \times \left(\delta\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \\ &\quad - \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N\right) \times \delta\underline{v}^N + \delta\underline{g}_{Mdl}^N \\ &\quad + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta h + C_2 \, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^N \end{aligned} \tag{12.3.4-39}$$

$$F(h) = 2 \quad \text{For } \; h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For } \; h < 0$$

(Continued)

$$\dot{\underline{\varepsilon}}^N = - \underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N$$

$$\delta\dot{h} = \underline{u}_{ZN}^N \cdot \delta\underline{v}^N - C_3 \left(\delta h - \delta h_{Prsr}\right)$$

$$\delta\dot{e}_{vc_3} = C_1 \left(\delta h - \delta h_{Prsr}\right)$$

$$\delta\underline{\omega}_{EN}^N = \delta\rho_{ZN}\, \underline{u}_{ZN}^N + \frac{1}{r_l} \left(\underline{u}_{ZN}^N \times \delta\underline{v}^N\right) - \frac{1}{r_l^2} \left(\underline{u}_{ZN}^N \times \underline{v}^N\right) \delta h \qquad \text{(12.3.4-39) (Continued)}$$

$$\delta\rho_{ZN} = 0 \qquad \text{For Wander Azimuth Implementation}$$

$$\delta\rho_{ZN} = - \left(\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \cdot \underline{u}_{ZN}^N \qquad \text{For Free Azimuth Implementation}$$

12.3.5 MIXED E AND N FRAME DEFINED ERROR EQUATION SET WRITTEN IN THE N FRAME

Equations (12.3.3-6) or (12.3.4-39) represent navigation error rate equations in N Frame coordinates that characterize the errors in navigation equation set (12.1.3-7) or (12.1.2-6). The attitude, velocity, position error parameters for these error equations are, respectively, $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ (for Equations (12.3.3-6)) and $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh (for Equations (12.3.4-39)). Sections 12.2.1 - 12.2.3 show the analytical equivalency between the two error parameter sets. Once a particular error rate equation set is selected (and integrated) the integrated result can be converted to the other error forms using Sections 12.2.1 - 12.2.3. This procedure underlies a fundamental principle in error parameter selection for the error rate integration process (as discussed in Section 12.2.5); namely that it is not necessary that the error parameters chosen for error rate equation integration directly reflect the actual navigation equations implemented in the strapdown system computer (e.g., Equation set (12.1.3-7) or (12.1.2-6)). The identical result is obtained by direct integration of a particular set of error rate equations or by selection of an alternate set and then converting the integrated result to generate the desired error parameters.

Based on the previous discussion, it is also possible to choose error parameters for integration that are mixed combinations of the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ and $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh parameters. This section analyzes the process by which such a set of error equations can be developed using the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ or $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh rate equations as a starting point. For example, let us consider the development of a self-consistent set of error rate equations for the $\underline{\gamma}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ error parameters. This can be achieved by modifying the $\dot{\underline{\gamma}}^N$ expression from the $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh equation set and $\delta\dot{\underline{V}}^N$ from the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ equation set to be functions of $\underline{\gamma}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ (Note - In the process, it will also be found that ε_{ZN}, the Z component of $\underline{\varepsilon}^N$, is required for the $\underline{\gamma}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ set). The $\delta\dot{\underline{R}}^N$ expression from the $\underline{\gamma}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ equation set is already compatible with the mixed $\underline{\gamma}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ error equation set.

We begin by first defining the equivalency between $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh and $\delta\underline{V}^N$, $\delta\underline{R}^N$ from Section 12.2.2 Equation (12.2.2-5) and Section 12.2.3 Equations (12.2.3-20), (12.2.3-23) and (12.2.3-25):

$$\delta\underline{v}^N = \delta\underline{V}^N - \underline{\varepsilon}^N \times \underline{v}^N \tag{12.3.5-1}$$

$$\underline{\varepsilon}^N = \frac{1}{R}\left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) + \varepsilon_{ZN}\, \underline{u}_{ZN}^N \tag{12.3.5-2}$$

$$\delta h \approx \delta R \tag{12.3.5-3}$$

$$\delta R = \underline{u}_{ZN}^N \cdot \delta\underline{R}^N \tag{12.3.5-4}$$

Let us now substitute (12.3.5-2) for $\underline{\varepsilon}^N$ into the Equations (12.3.4-39) $\dot{\underline{\gamma}}^N$ expression to obtain the equivalent equation in terms of the $\delta\underline{R}^N$ parameter:

$$\begin{aligned}\dot{\underline{\gamma}}^N = &- C_B^N\, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \frac{1}{R}\, \underline{\omega}_{IE}^N \times \left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) \\ &+ \left(\underline{\omega}_{IE}^N \times \underline{u}_{ZN}^N\right) \varepsilon_{ZN} + \delta\underline{\omega}_{EN}^N\end{aligned} \tag{12.3.5-5}$$

The $\delta\underline{\omega}_{EN}^N$ term in (12.3.5-5) is obtained from Equations (12.3.4-39) by first using (12.3.5-3) to replace δh:

$$\delta\underline{\omega}_{EN}^N = \delta\rho_{ZN}\, \underline{u}_{ZN}^N + \frac{1}{r_l}\left(\underline{u}_{ZN}^N \times \delta\underline{v}^N\right) - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^N \times \underline{v}^N\right)\delta R \tag{12.3.5-6}$$

The $\underline{u}_{ZN}^N \times \delta\underline{v}^N$ term in (12.3.5-6) can be modified with Equation (12.3.5-1):

$$\underline{u}_{ZN}^N \times \delta\underline{v}^N = \underline{u}_{ZN}^N \times \delta\underline{V}^N - \underline{u}_{ZN}^N \times \left(\underline{\varepsilon}^N \times \underline{v}^N\right) \tag{12.3.5-7}$$

The $\underline{u}_{ZN}^N \times \left(\underline{\varepsilon}^N \times \underline{v}^N\right)$ term in (12.3.5-7) is expanded using the Equation (3.1.1-16) vector triple cross-product rule:

$$\underline{u}_{ZN}^N \times \left(\underline{\varepsilon}^N \times \underline{v}^N\right) = \underline{\varepsilon}^N\left(\underline{v}^N \cdot \underline{u}_{ZN}^N\right) - \underline{v}^N\left(\underline{\varepsilon}^N \cdot \underline{u}_{ZN}^N\right) = v_{ZN}\, \underline{\varepsilon}^N - \varepsilon_{ZN}\, \underline{v}^N \tag{12.3.5-8}$$

where

v_{ZN} = Vertical component of $\underline{v}^N$ (i.e., along $\underline{u}^N_{ZN}$).

or with $\underline{v}^N$ equated to the sum of its vertical and horizontal components, substitution of (12.3.5-2) for $\underline{\varepsilon}^N$, and approximating R by r_l as will be shown subsequently in Section 12.3.6.1, Equation (12.3.6.1-26):

$$\underline{u}^N_{ZN} \times \left(\underline{\varepsilon}^N \times \underline{v}^N\right) = v_{ZN} \frac{1}{r_l}\left(\underline{u}^N_{ZN} \times \delta\underline{R}^N\right) - \underline{v}^N_H \, \varepsilon_{ZN} \qquad (12.3.5\text{-}9)$$

where

$\underline{v}^N_H$ = Horizontal component of $\underline{v}^N$ (i.e., perpendicular to $\underline{u}^N_{ZN}$).

With (12.3.5-7) and (12.3.5-9), the Equation (12.3.5-6) $\delta\underline{\omega}^N_{EN}$ expression becomes the form desired for $\dot{\underline{\gamma}}^N$ Equation (12.3.5-5):

$$\begin{aligned}\delta\underline{\omega}^N_{EN} = {} & \delta\rho_{ZN}\, \underline{u}^N_{ZN} + \frac{1}{r_l}\left(\underline{u}^N_{ZN} \times \delta\underline{V}^N\right) - v_{ZN}\frac{1}{r_l^2}\left(\underline{u}^N_{ZN} \times \delta\underline{R}^N\right) \\ & + \frac{1}{r_l}\underline{v}^N_H \, \varepsilon_{ZN} - \frac{1}{r_l^2}\left(\underline{u}^N_{ZN} \times \underline{v}^N\right)\delta R\end{aligned} \qquad (12.3.5\text{-}10)$$

The $\delta\rho_{ZN}$ term in (12.3.5-10) is provided in terms of $\delta\underline{R}^N$ by Equations (12.2.4-14) and (12.2.4-26):

$$\begin{aligned}&\delta\rho_{ZN} = 0 \qquad \text{For Wander Azimuth Implementation} \\ &\delta\rho_{ZN} = -\frac{1}{R}\,\underline{\omega}^N_{IE_H} \cdot \delta\underline{R}^N_H \qquad \text{For Free Azimuth Implementation}\end{aligned} \qquad (12.3.5\text{-}11)$$

The ε_{ZN} term in (12.3.5-5) and (12.3.5-10) (i.e., the vertical component of $\underline{\varepsilon}^N$) is obtained in terms of $\delta\underline{R}^N$ from the Equations (12.3.4-39) $\underline{\varepsilon}^N$ rate expression:

$$\dot{\underline{\varepsilon}}^N = -\underline{\omega}^N_{EN} \times \underline{\varepsilon}^N + \delta\underline{\omega}^N_{EN} \qquad (12.3.5\text{-}12)$$

Substituting (12.3.5-2) for $\underline{\varepsilon}^N$ in (12.3.5-12) gives:

$$\dot{\underline{\varepsilon}}^N = -\frac{1}{R}\,\underline{\omega}^N_{EN} \times \left(\underline{u}^N_{ZN} \times \delta\underline{R}^N\right) - \left(\underline{\omega}^N_{EN} \times \underline{u}^N_{ZN}\right)\varepsilon_{ZN} + \delta\underline{\omega}^N_{EN} \qquad (12.3.5\text{-}13)$$

or after applying the vector triple product rule (Equation (3.1.1-16)):

$$\dot{\underline{\varepsilon}}^N = -\frac{1}{R}\left[\left(\underline{\omega}_{EN}^N \cdot \delta\underline{R}^N\right)\underline{u}_{ZN}^N - \left(\underline{\omega}_{EN}^N \cdot \underline{u}_{ZN}^N\right)\delta\underline{R}^N\right] - \left(\underline{\omega}_{EN}^N \times \underline{u}_{ZN}^N\right)\varepsilon_{ZN} + \delta\underline{\omega}_{EN}^N \qquad (12.3.5\text{-}14)$$

The ε_{ZN} component of $\underline{\varepsilon}^N$ is defined by, and from Equation (12.3.5-2), given by:

$$\varepsilon_{ZN} = \underline{u}_{ZN}^N \cdot \underline{\varepsilon}^N \qquad (12.3.5\text{-}15)$$

Taking the time derivative of (12.3.5-15) (with $\underline{u}_{ZN}^N$ recognized to be constant) and substitution of $\dot{\underline{\varepsilon}}^N$ from (12.3.5-14) then yields:

$$\dot{\varepsilon}_{ZN} = -\frac{1}{R}\left[\left(\underline{\omega}_{EN}^N \cdot \delta\underline{R}^N\right) - \left(\underline{\omega}_{EN}^N \cdot \underline{u}_{ZN}^N\right)\left(\underline{u}_{ZN}^N \cdot \delta\underline{R}^N\right)\right] + \underline{u}_{ZN}^N \cdot \delta\underline{\omega}_{EN}^N \qquad (12.3.5\text{-}16)$$

The $\underline{u}_{ZN}^N \cdot \delta\underline{\omega}_{EN}^N$ term in (12.3.5-16) is from Equation (12.3.5-10):

$$\underline{u}_{ZN}^N \cdot \delta\underline{\omega}_{EN}^N = \delta\rho_{ZN} \qquad (12.3.5\text{-}17)$$

with $\delta\rho_{ZN}$ from Equations (12.3.5-11).

The remaining terms in (12.3.5-16) can be simplified by incorporating the rearranged N Frame version of Equation (12.2.4-6):

$$\delta\underline{R}^N = \delta\underline{R}_H^N + \delta R\, \underline{u}_{ZN}^N \qquad (12.3.5\text{-}18)$$

and the $\underline{\omega}_{EN}^N$ expression from Equations (12.1.2-6) in the compressed form:

$$\underline{\omega}_{EN}^N = \underline{\omega}_{EN_H}^N + \rho_{ZN}\, \underline{u}_{ZN}^N \qquad (12.3.5\text{-}19)$$

where

$$\underline{\omega}_{EN_H}^N = \text{Horizontal portion of } \underline{\omega}_{EN}^N.$$

With (12.3.5-18) and (12.3.5-19), particular terms in (12.3.5-16) become:

$$\begin{aligned} &\underline{\omega}_{EN}^N \cdot \delta\underline{R}^N = \rho_{ZN}\,\delta R + \underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N \\ &\left(\underline{\omega}_{EN}^N \cdot \underline{u}_{ZN}^N\right)\left(\underline{u}_{ZN}^N \cdot \delta\underline{R}^N\right) = \rho_{ZN}\,\delta R \end{aligned} \qquad (12.3.5\text{-}20)$$

Substituting (12.3.5-17) and (12.3.5-20) into (12.3.5-16) then yields the desired form of the $\dot{\varepsilon}_{ZN}$ equation:

$$\dot{\varepsilon}_{ZN} = - \frac{1}{R} \left(\underline{\omega}^N_{EN_H} \cdot \delta \underline{R}^N_H \right) + \delta\rho_{ZN} \tag{12.3.5-21}$$

Conversion of the Equations (12.3.3-6) $\delta\dot{\underline{V}}^N$ velocity error rate expression to be in terms of $\underline{\gamma}^N$, $\delta\underline{R}^N$ is facilitated by first approximating the $\underline{g}^N_P$ plumb-bob gravity term in (12.1.2-6) (as we did in Equation (12.2.4-2)) by:

$$\underline{g}^N_P \approx - g \, \underline{u}^N_{ZN} \tag{12.3.5-22}$$

where

g = Magnitude of gravity at position $\underline{R}^N$.

With (12.3.5-22), we now define $\underline{a}^N_{SF}$ as the sum of a component to counteract gravity plus a remainder (excess) attributable to maneuvering:

$$\underline{a}^N_{SF} = - \underline{g}^N_P + \Delta\underline{a}^N_{SF} \approx g \, \underline{u}^N_{ZN} + \Delta\underline{a}^N_{SF} \tag{12.3.5-23}$$

where

$\Delta\underline{a}^N_{SF}$ = Excess portion of $\underline{a}^N_{SF}$ for maneuvering. The $g \, \underline{u}^N_{ZN}$ term in (12.3.5-23) is the upward specific force acceleration contribution from $\underline{a}^N_{SF}$ that counteracts the plumb-bob gravitational acceleration $\underline{g}^N_P$.

We also make use of equivalency Equation (12.2.1-17):

$$\underline{\psi}^N = \underline{\gamma}^N - \underline{\varepsilon}^N \tag{12.3.5-24}$$

With (12.3.5-23) and (12.3.5-24), the $\underline{a}^N_{SF} \times \underline{\psi}^N$ term in the Equations (12.3.3-6) $\delta\dot{\underline{V}}^N$ velocity error rate expression becomes:

$$\underline{a}^N_{SF} \times \underline{\psi}^N = \underline{a}^N_{SF} \times \underline{\gamma}^N - \Delta\underline{a}^N_{SF} \times \underline{\varepsilon}^N - g \, \underline{u}^N_{ZN} \times \underline{\varepsilon}^N \tag{12.3.5-25}$$

The $g \, \underline{u}^N_{ZN} \times \underline{\varepsilon}^N$ term in (12.3.5-25) is with (12.3.5-2), (12.3.5-4), (12.3.5-18) and the (3.1.1-16) vector triple cross-product rule:

$$
\begin{aligned}
g\,\underline{u}_{ZN}^N \times \underline{\varepsilon}^N &= \frac{g}{R}\,\underline{u}_{ZN}^N \times \left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) \\
&= \frac{g}{R}\left[\left(\underline{u}_{ZN}^N \cdot \delta\underline{R}^N\right)\underline{u}_{ZN}^N - \delta\underline{R}^N\right] = -\frac{g}{R}\,\delta\underline{R}_H^N
\end{aligned}
\tag{12.3.5-26}
$$

We finally substitute (12.3.5-25) with (12.3.5-26) and (12.3.5-2) into the Equations (12.3.3-6) $\delta\dot{\underline{V}}^N$ velocity error rate expression to obtain the desired form:

$$
\begin{aligned}
\delta\dot{\underline{V}}^N &= C_B^N\,\delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N - \Delta\underline{a}_{SF}^N \times \left[\frac{1}{R}\left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) + \varepsilon_{ZN}\,\underline{u}_{ZN}^N\right] \\
&\quad - \left(\underline{\omega}_{IE}^N + \underline{\omega}_{IN}^N\right) \times \delta\underline{V}^N + \delta\underline{g}_{Mdl}^N + \left[\left(F(h)\,\frac{g}{R} - C_2\right)\delta R + C_2\,\delta h_{Prsr} - \delta e_{vc_3}\right]\underline{u}_{ZN}^N \\
F(h) &= 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0
\end{aligned}
\tag{12.3.5-27}
$$

The δe_{vc_3} term for Equation (12.3.5-27) and the $\delta\dot{\underline{R}}^N$ equation are obtained unchanged from Equations (12.3.3-6):

$$
\begin{aligned}
\delta\dot{\underline{R}}^N &= \delta\underline{V}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N - C_3\left(\delta R - \delta h_{Prsr}\right)\underline{u}_{ZN}^N \\
\delta\dot{e}_{vc_3} &= C_1\left(\delta R - \delta h_{Prsr}\right)
\end{aligned}
\tag{12.3.5-28}
$$

In summary, the navigation error rate equations in terms of the $\underline{\gamma}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ parameters are given by Equations (12.3.5-4), (12.3.5-5), (12.3.5-10), (12.3.5-11), (12.3.5-18) rearranged, (12.3.5-19) rearranged, (12.3.5-21), (12.3.5-23) rearranged, (12.3.5-27) and (12.3.5-28) repeated below for easy reference.

$$
\begin{aligned}
\dot{\underline{\gamma}}^N &= -C_B^N\,\delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \frac{1}{R}\,\underline{\omega}_{IE}^N \times \left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) + \left(\underline{\omega}_{IE}^N \times \underline{u}_{ZN}^N\right)\varepsilon_{ZN} + \delta\underline{\omega}_{EN}^N \\
\dot{\varepsilon}_{ZN} &= -\frac{1}{R}\left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right) + \delta\rho_{ZN} \\
\delta\dot{\underline{V}}^N &= C_B^N\,\delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N - \Delta\underline{a}_{SF}^N \times \left[\frac{1}{R}\left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) + \varepsilon_{ZN}\,\underline{u}_{ZN}^N\right] \\
&\quad - \left(\underline{\omega}_{IE}^N + \underline{\omega}_{IN}^N\right) \times \delta\underline{V}^N + \delta\underline{g}_{Mdl}^N + \left[\left(F(h)\,\frac{g}{R} - C_2\right)\delta R + C_2\,\delta h_{Prsr} - \delta e_{vc_3}\right]\underline{u}_{ZN}^N \\
F(h) &= 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \\
\delta\dot{e}_{vc_3} &= C_1\left(\delta R - \delta h_{Prsr}\right)
\end{aligned}
\tag{12.3.5-29}
$$

(Continued)

$$\delta\underline{\dot{R}}^N = \delta\underline{V}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N - C_3\left(\delta R - \delta h_{Prsr}\right)\underline{u}_{ZN}^N$$

$$\delta\underline{\omega}_{EN}^N = \delta\rho_{ZN}\,\underline{u}_{ZN}^N + \frac{1}{r_l}\left(\underline{u}_{ZN}^N \times \delta\underline{V}^N\right) - v_{ZN}\,\frac{1}{r_l^2}\left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right)$$

$$+ \frac{1}{r_l}\,\underline{v}_H^N\,\varepsilon_{ZN} - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^N \times \underline{v}^N\right)\delta R \qquad \begin{array}{c}(12.3.5\text{-}29)\\ (\text{Continued})\end{array}$$

$\delta\rho_{ZN} = 0$ For Wander Azimuth Implementation

$\delta\rho_{ZN} = -\frac{1}{R}\,\underline{\omega}_{IE_H}^N \cdot \delta\underline{R}_H^N$ For Free Azimuth Implementation

$$\delta R = \underline{u}_{ZN}^N \cdot \delta\underline{R}^N$$

$$\delta\underline{R}_H^N = \delta\underline{R}^N - \delta R\,\underline{u}_{ZN}^N$$

$$\underline{\omega}_{EN_H}^N = \underline{\omega}_{EN}^N - \rho_{ZN}\,\underline{u}_{ZN}^N$$

$$\Delta\underline{a}_{SF}^N = \underline{a}_{SF}^N + \underline{g}_P^N$$

12.3.6 EQUIVALENCIES BETWEEN E FRAME AND N FRAME DEFINED ERROR PARAMETER DIFFERENTIAL EQUATIONS

The rate equations developed in Section 12.3.3 for the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ navigation error parameters could also have been derived from the Section 12.3.4 $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh error parameter rate equations using the Section 12.2.1 - 12.2.3 error parameter equivalencies. Similarly, the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ error rate equations can be converted directly to the $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh error rate equations if the vertical component of $\underline{\varepsilon}^N$ is defined independently. Section 12.3.6.1 discusses the former conversion process while Section 12.3.6.2 discusses the latter conversion process. These sections can be viewed as an exercise or as a verification that the final results match the equivalent results obtained in Sections 12.3.3 and 12.3.4 .

12.3.6.1 E FRAME DEFINED ERROR PARAMETER RATE EQUATIONS FROM N FRAME DEFINED ERROR PARAMETER RATE EQUATIONS

In this section we derive the time rate equations for the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ navigation error parameters beginning with the Section 12.3.4 differential equations for $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh

(summarized in Equations (12.3.4-39)) and incorporating the Section 12.2.1 - 12.2.3 equivalencies between $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh and $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$.

We begin with the equivalency relationship between $\underline{\psi}^N$, $\underline{\gamma}^N$ and $\underline{\varepsilon}^N$ from Equation (12.2.1-17):

$$\underline{\psi}^N = \underline{\gamma}^N - \underline{\varepsilon}^N \tag{12.3.6.1-1}$$

Taking the time derivative of (12.3.6.1-1):

$$\dot{\underline{\psi}}^N = \dot{\underline{\gamma}}^N - \dot{\underline{\varepsilon}}^N \tag{12.3.6.1-2}$$

The $\dot{\underline{\gamma}}^N$ term in (12.3.6.1-2) is from Equations (12.3.4-39):

$$\dot{\underline{\gamma}}^N = - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \tag{12.3.6.1-3}$$

The $\underline{\omega}_{IN}^N$ term in (12.3.6.1-3) equals earth rate $\underline{\omega}_{IE}^N$ plus transport rate $\underline{\omega}_{EN}^N$:

$$\underline{\omega}_{IN}^N = \underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N \tag{12.3.6.1-4}$$

Substituting (12.3.6.1-1) and (12.3.6.1-4) into (12.3.6.1-3) yields:

$$\begin{aligned}
\dot{\underline{\gamma}}^N &= - C_B^N \, \delta\underline{\omega}_{IB}^B - \left(\underline{\omega}_{IE}^N + \underline{\omega}_{EN}^N\right) \times \underline{\gamma}^N + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \\
&= - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{EN}^N \times \underline{\gamma}^N - \underline{\omega}_{IE}^N \times \left(\underline{\gamma}^N - \underline{\varepsilon}^N\right) + \delta\underline{\omega}_{EN}^N \\
&= - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{EN}^N \times \underline{\gamma}^N - \underline{\omega}_{IE}^N \times \underline{\psi}^N + \delta\underline{\omega}_{EN}^N
\end{aligned} \tag{12.3.6.1-5}$$

The $\dot{\underline{\varepsilon}}^N$ term in (12.3.6.1-2) is from Equations (12.3.4-39):

$$\dot{\underline{\varepsilon}}^N = - \underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \tag{12.3.6.1-6}$$

Substituting (12.3.6.1-5) and (12.3.6.1-6) into (12.3.6.1-2) gives:

$$\dot{\underline{\psi}}^N = \dot{\underline{\gamma}}^N - \dot{\underline{\varepsilon}}^N = - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{EN}^N \times \left(\underline{\gamma}^N - \underline{\varepsilon}^N\right) - \underline{\omega}_{IE}^N \times \underline{\psi}^N \tag{12.3.6.1-7}$$

or with (12.3.6.1-1) and (12.3.6.1-4), the final result:

$$\dot{\underline{\psi}}^N = - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\psi}^N \tag{12.3.6.1-8}$$

Equation (12.3.6.1-8) matches the result for $\dot{\underline{\psi}}^N$ in Equations (12.3.3-6).

The $\delta\dot{\underline{V}}^N$ equation is derived from $\delta\dot{\underline{v}}^N$ in Equations (12.3.4-39) beginning with the Equation (12.2.2-5) equivalency relation:

$$\delta\underline{V}^N = \delta\underline{v}^N + \underline{\varepsilon}^N \times \underline{v}^N \qquad (12.3.6.1\text{-}9)$$

The time derivative of (12.3.6.1-9) is:

$$\delta\dot{\underline{V}}^N = \delta\dot{\underline{v}}^N + \dot{\underline{\varepsilon}}^N \times \underline{v}^N + \underline{\varepsilon}^N \times \dot{\underline{v}}^N \qquad (12.3.6.1\text{-}10)$$

The $\delta\dot{\underline{v}}^N$ term in (12.3.6.1-10) is from Equations (12.3.4-39):

$$\begin{aligned} \delta\dot{\underline{v}}^N = \; & C_B^N \, \delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N + \underline{v}^N \times \left(\delta\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \\ & - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \delta\underline{v}^N + \delta\underline{g}_{Mdl}^N \\ & + \left[\left(F(h)\,\frac{g}{R} - C_2\right)\delta h + C_2\,\delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^N \end{aligned} \qquad (12.3.6.1\text{-}11)$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

The δh term in (12.3.6.1-11) is from Equation (12.2.3-23):

$$\delta h = \delta R \qquad (12.3.6.1\text{-}12)$$

The $\dot{\underline{\varepsilon}}^N$ term in (12.3.6.1-10) is provided by Equation (12.3.6.1-6). The $\dot{\underline{v}}^N$ term in Equation (12.3.6.1-10) is from Equations (12.1.2-6):

$$\dot{\underline{v}}^N = C_B^N \, \underline{a}_{SF}^B + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \underline{v}^N - e_{vc_1}\,\underline{u}_{ZN}^N \qquad (12.3.6.1\text{-}13)$$

with

$$e_{vc_1} = e_{vc_3} + C_2\left(h - h_{Prsr}\right) \qquad (12.3.6.1\text{-}14)$$

Substituting (12.3.6.1-6), (12.3.6.1-9) rearranged, and (12.3.6.1-11) - (12.3.6.1-14) into (12.3.6.1-10) then obtains for $\delta\dot{\underline{V}}^N$:

$$
\begin{aligned}
\delta \dot{\underline{V}}^N = \; & C_B^N \, \delta \underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N + \underline{v}^N \times \left(\delta \underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N \right) \\
& - \left(\underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \right) \times \left(\delta \underline{V}^N - \underline{\varepsilon}^N \times \underline{v}^N \right) + \delta \underline{g}_{Mdl}^N \\
& + \left[\left(F(h) \, \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^N + \left(- \, \underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \delta \underline{\omega}_{EN}^N \right) \times \underline{v}^N \\
& + \underline{\varepsilon}^N \times \left[C_B^N \, \underline{a}_{SF}^B + \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \right) \times \underline{v}^N - e_{vc_1} \, \underline{u}_{ZN}^N \right] \\
= \; & C_B^N \, \delta \underline{a}_{SF}^B + \underline{a}_{SF}^N \times \left(\underline{\gamma}^N - \underline{\varepsilon}^N \right) + \underline{\varepsilon}^N \times \underline{g}_P^N - \left(\underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \right) \times \delta \underline{V}^N + \delta \underline{g}_{Mdl}^N \\
& + \left[\left(F(h) \, \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^N - e_{vc_1} \left(\underline{\varepsilon}^N \times \underline{u}_{ZN}^N \right) \\
& + \underline{v}^N \times \left[\left(\underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \right) \times \underline{\varepsilon}^N \right] + \left(\underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \right) \times \left(\underline{\varepsilon}^N \times \underline{v}^N \right) \\
& - \underline{\varepsilon}^N \times \left[\left(\underline{\omega}_{EN}^N + 2 \, \underline{\omega}_{IE}^N \right) \times \underline{v}^N \right]
\end{aligned}
\tag{12.3.6.1-15}
$$

$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$

We apply Equation (12.3.6.1-1) for the $\left(\underline{\gamma}^N - \underline{\varepsilon}^N \right)$ term in (12.3.6.1-15). The $\underline{\varepsilon}^N \times \underline{u}_{ZN}^N$ term in (12.3.6.1-15) is obtained from Equations (12.2.3-16) and (12.2.4-6):

$$
\delta \underline{R}^N = R \left(\underline{\varepsilon}^N \times \underline{u}_{ZN}^N \right) + \delta R \, \underline{u}_{ZN}^N = \delta \underline{R}_H^N + \delta R \, \underline{u}_{ZN}^N \tag{12.3.6.1-16}
$$

which shows that:

$$
\underline{\varepsilon}^N \times \underline{u}_{ZN}^N = \frac{1}{R} \, \delta \underline{R}_H^N \tag{12.3.6.1-17}
$$

As in Section 12.3.5 Equation (12.3.5-22) we also approximate the $\underline{g}_P^N$ plumb-bob gravity term in (12.3.6.1-15) as:

$$
\underline{g}_P^N \approx - \, g \, \underline{u}_{ZN}^N \tag{12.3.6.1-18}
$$

which, with (12.3.6.1-17), gives for $\underline{\varepsilon}^N \times \underline{g}_P^N$ in Equation (12.3.6.1-15):

$$
\underline{\varepsilon}^N \times \underline{g}_P^N = - \, \frac{g}{R} \, \delta \underline{R}_H^N \tag{12.3.6.1-19}
$$

Finally, the last three terms in Equation (12.3.6.1-15) can be shown to sum to zero by application of generalized Equations (3.1.1-15) and (3.1.1-21):

$$
\begin{aligned}
&\underline{v}^N \times \left[\left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \times \underline{\varepsilon}^N \right] + \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \times \left(\underline{\varepsilon}^N \times \underline{v}^N \right) \\
&\qquad - \underline{\varepsilon}^N \times \left[\left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \times \underline{v}^N \right] \\
&= \underline{v}^N \times \left[\left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \times \underline{\varepsilon}^N \right] - \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \times \left(\underline{v}^N \times \underline{\varepsilon}^N \right) \\
&\qquad - \left[\underline{v}^N \times \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \right] \times \underline{\varepsilon}^N \\
&= \left(\underline{v}^N \times \right) \left[\left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \times \right] \underline{\varepsilon}^N - \left[\left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \times \right] \left(\underline{v}^N \times \right) \underline{\varepsilon}^N \\
&\qquad - \left(\left[\underline{v}^N \times \left(\underline{\omega}_{EN}^N + 2\, \underline{\omega}_{IE}^N \right) \right] \times \right) \underline{\varepsilon}^N = 0
\end{aligned}
\tag{12.3.6.1-20}
$$

Substituting (12.3.6.1-1), (12.3.6.1-4), (12.3.6.1-17), (12.3.6.1-19) and (12.3.6.1-20) into (12.3.6.1-15) then yields for $\delta \dot{\underline{V}}^N$:

$$
\begin{aligned}
\delta \dot{\underline{V}}^N &= C_B^N\, \delta \underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\psi}^N - \frac{g}{R}\, \delta \underline{R}_H^N - \left(\underline{\omega}_{IE}^N + \underline{\omega}_{IN}^N \right) \times \delta \underline{V}^N + \delta \underline{g}_{Mdl}^N \\
&\quad + \left[\left(F(h)\, \frac{g}{R} - C_2 \right) \delta R + C_2\, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^N - e_{vc_1}\, \frac{1}{R}\, \delta \underline{R}_H^N
\end{aligned}
\tag{12.3.6.1-21}
$$

$$
F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0
$$

Equation (12.3.6.1-21) is identical to the $\delta \dot{\underline{V}}^N$ expression in Equations (12.3.3-6) except for the $e_{vc_1} \frac{1}{R}\, \delta \underline{R}_H^N$ term. Recall, however, that $\delta \dot{\underline{V}}^E$ from Equations (12.3.2-43) was used to derive $\delta \dot{\underline{V}}^N$ in Equations (12.3.3-6), and that in the derivation of $\delta \dot{\underline{V}}^E$, a term $e_{vc_1} \frac{1}{R}\, \delta \underline{R}_H^E$ was dropped as negligible. If the $e_{vc_1} \frac{1}{R}\, \delta \underline{R}_H^E$ term was carried, $\delta \dot{\underline{V}}^N$ in Equations (12.3.3-6) would identically match the Equation (12.3.6.1-21) result.

For completeness, we also drop the $e_{vc_1} \frac{1}{R}\, \delta \underline{R}_H^N$ term in (12.3.6.1-21) as negligible to obtain a final result which identically matches $\delta \dot{\underline{V}}^N$ in Equations (12.3.3-6):

$$
\begin{aligned}
\delta \dot{\underline{V}}^N &= C_B^N\, \delta \underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\psi}^N - \frac{g}{R}\, \delta \underline{R}_H^N - \left(\underline{\omega}_{IE}^N + \underline{\omega}_{IN}^N \right) \times \delta \underline{V}^N + \delta \underline{g}_{Mdl}^N \\
&\quad + \left[\left(F(h)\, \frac{g}{R} - C_2 \right) \delta R + C_2\, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^N
\end{aligned}
\tag{12.3.6.1-22}
$$

$$
F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0
$$

Development of the $\delta\dot{\underline{R}}^N$ equation from $\dot{\underline{\varepsilon}}^N$ and $\delta\dot{h}$ is somewhat more involved. We first develop the horizontal component of the $\delta\underline{R}$ time derivative in the E Frame, transform the result to the N Frame and add the vertical component of the $\delta\underline{R}$ N Frame time derivative to obtain $\delta\dot{\underline{R}}^N$. We begin with Equation (12.3.6.1-17) transformed to the E Frame:

$$\underline{\varepsilon}^E \times \underline{u}^E_{ZN} = \frac{1}{R}\,\delta\underline{R}^E_H \tag{12.3.6.1-23}$$

The time derivative of (12.3.6.1-23) is with (12.3.6.1-23):

$$\dot{\underline{\varepsilon}}^E \times \underline{u}^E_{ZN} + \underline{\varepsilon}^E \times \dot{\underline{u}}^E_{ZN} = \frac{1}{R}\,\delta\dot{\underline{R}}^E_H - \frac{\dot{R}}{R^2}\,\delta\underline{R}^E_H = \frac{1}{R}\,\delta\dot{\underline{R}}^E_H - \frac{\dot{R}}{R}\left(\underline{\varepsilon}^E \times \underline{u}^E_{ZN}\right) \tag{12.3.6.1-24}$$

Applying generalized Equation (3.3.1-2) to $\dot{\underline{u}}^E_{ZN}$ in (12.3.6.1-24) while recognizing that $\underline{u}^E_{ZN}$ is a constant unit vector in the N Frame gives:

$$\dot{\underline{u}}^E_{ZN} = \underline{\omega}^E_{EN} \times \underline{u}^E_{ZN} \tag{12.3.6.1-25}$$

The $\underline{\omega}^E_{EN}$ term in (12.3.6.1-25) is, from navigation Equations (12.1.2-6), a function of the radius of curvature term r_l. From Equations (12.1.2-6), r_l can be simplified for error analysis purposes as follows:

$$r_l = r_{ls} + h = \left[1 + 2\left(2\,u^2_{ZN_{YE}} - 1\right)e\right]R_S + h \approx R_S + h \approx R \tag{12.3.6.1-26}$$

With (12.3.6.1-26) for r_l, the $\underline{\omega}^E_{EN}$ term is then obtained from Equations (12.1.2-6) transformed to the E Frame, with the ∂G_C term dropped as negligible for error analysis purposes:

$$\underline{\omega}^E_{EN} \approx \rho_{ZN}\,\underline{u}^E_{ZN} + \frac{1}{R}\left(\underline{u}^E_{ZN} \times \underline{v}^E\right) \tag{12.3.6.1-27}$$

The velocity vector $\underline{v}^E$ in (12.3.6.1-27) can be decomposed into its vertical (along $\underline{u}^E_{ZN}$) and horizontal (perpendicular to $\underline{u}^E_{ZN}$) components:

$$\underline{v}^E = \underline{v}^E_H + \left(\underline{u}^E_{ZN} \cdot \underline{v}^E\right)\underline{u}^E_{ZN} \tag{12.3.6.1-28}$$

where

$\underline{v}^E_H$ = Horizontal component of $\underline{v}^E$.

With (12.3.6.1-28), Equation (12.3.6.1-27) is equivalently:

$$\underline{\omega}_{EN}^{E} \approx \rho_{ZN}\, \underline{u}_{ZN}^{E} + \frac{1}{R}\left(\underline{u}_{ZN}^{E} \times \underline{v}_{H}^{E}\right) \tag{12.3.6.1-29}$$

We also see then upon application of the general Equation (3.1.1-16) triple vector cross-product rule that:

$$\underline{\omega}_{EN}^{E} \times \underline{u}_{ZN}^{E} = \frac{1}{R}\left(\underline{u}_{ZN}^{E} \times \underline{v}_{H}^{E}\right) \times \underline{u}_{ZN}^{E} = \frac{1}{R}\,\underline{v}_{H}^{E} \tag{12.3.6.1-30}$$

Thus, with (12.3.6.1-30) and (12.3.6.1-25), $\dot{\underline{u}}_{ZN}^{E}$ in Equation (12.3.6.1-24) is given by:

$$\dot{\underline{u}}_{ZN}^{E} = \frac{1}{R}\,\underline{v}_{H}^{E} \tag{12.3.6.1-31}$$

The $\dot{R}$ term in (12.3.6.1-24) is determined from (12.1.3-6):

$$R^2 = \underline{R}^{E} \cdot \underline{R}^{E} \tag{12.3.6.1-32}$$

Taking the derivative of (12.3.6.1-32) yields upon rearrangement:

$$\dot{R} = \frac{1}{R}\,\underline{R}^{E} \cdot \dot{\underline{R}}^{E} \tag{12.3.6.1-33}$$

which with $\dot{\underline{R}}^{E}$ identified as $\underline{v}^{E}$ from (4.3-1) and application of (12.2.3-6) for $\underline{R}^{E}$ gives:

$$\dot{R} = \underline{u}_{ZN}^{E} \cdot \underline{v}^{E} \tag{12.3.6.1-34}$$

We now substitute (12.3.6.1-31) and (12.3.6.1-34) into (12.3.6.1-24) to obtain after rearrangement for $\delta\dot{\underline{R}}_{H}^{E}$:

$$\begin{aligned}\delta\dot{\underline{R}}_{H}^{E} &= R\left(\dot{\underline{\varepsilon}}^{E} \times \underline{u}_{ZN}^{E}\right) + \left(\underline{u}_{ZN}^{E} \cdot \underline{v}^{E}\right)\left(\underline{\varepsilon}^{E} \times \underline{u}_{ZN}^{E}\right) + \underline{\varepsilon}^{E} \times \underline{v}_{H}^{E} \\ &= R\left(\dot{\underline{\varepsilon}}^{E} \times \underline{u}_{ZN}^{E}\right) + \underline{\varepsilon}^{E} \times \left[\underline{v}_{H}^{E} + \left(\underline{u}_{ZN}^{E} \cdot \underline{v}^{E}\right)\underline{u}_{ZN}^{E}\right]\end{aligned} \tag{12.3.6.1-35}$$

or with (12.3.6.1-28):

$$\delta\dot{\underline{R}}_{H}^{E} = R\left(\dot{\underline{\varepsilon}}^{E} \times \underline{u}_{ZN}^{E}\right) + \underline{\varepsilon}^{E} \times \underline{v}^{E} \tag{12.3.6.1-36}$$

The $\dot{\underline{\varepsilon}}^E \times \underline{u}_{ZN}^E$ term in (12.3.6.1-36) is evaluated from the E Frame version of (12.3.6.1-6) using generalized vector transformation Equation (3.4-6):

$$\dot{\underline{\varepsilon}}^E = \delta\underline{\omega}_{EN}^E \tag{12.3.6.1-37}$$

From Equations (12.3.4-39), the E Frame version of $\delta\underline{\omega}_{EN}$ is with (12.3.6.1-26) for r_l and (12.3.6.1-12) for δh:

$$\delta\underline{\omega}_{EN}^E = \delta\rho_{ZN}\,\underline{u}_{ZN}^E + \frac{1}{R}\left(\underline{u}_{ZN}^E \times \delta\underline{v}^E\right) - \frac{1}{R^2}\left(\underline{u}_{ZN}^E \times \underline{v}^E\right)\delta R \tag{12.3.6.1-38}$$

Substituting (12.3.6.1-38) in (12.3.6.1-37) and taking the cross-product with $\underline{u}_{ZN}^E$ gives:

$$\dot{\underline{\varepsilon}}^E \times \underline{u}_{ZN}^E = \frac{1}{R}\left(\underline{u}_{ZN}^E \times \delta\underline{v}^E\right) \times \underline{u}_{ZN}^E - \frac{\delta R}{R}\,\frac{1}{R}\left(\underline{u}_{ZN}^E \times \underline{v}^E\right) \times \underline{u}_{ZN}^E \tag{12.3.6.1-39}$$

But, from the cross-product of (12.3.6.1-27) with $\underline{u}_{ZN}^E$ we see that:

$$\frac{1}{R}\left(\underline{u}_{ZN}^E \times \underline{v}^E\right) \times \underline{u}_{ZN}^E = \underline{\omega}_{EN}^E \times \underline{u}_{ZN}^E \tag{12.3.6.1-40}$$

Also, following the same logic leading to Equation (12.3.6.1-30), we can write:

$$\left(\underline{u}_{ZN}^E \times \delta\underline{v}^E\right) \times \underline{u}_{ZN}^E = \delta\underline{v}_H^E \tag{12.3.6.1-41}$$

where

$\delta\underline{v}_H^E$ = Horizontal component of $\delta\underline{v}^E$.

Substituting (12.3.6.1-39) with (12.3.6.1-40) and (12.3.6.1-41) into (12.3.6.1-36) then yields:

$$\delta\dot{\underline{R}}_H^E = \delta\underline{v}_H^E + \underline{\varepsilon}^E \times \underline{v}^E - \delta R\left(\underline{\omega}_{EN}^E \times \underline{u}_{ZN}^E\right) \tag{12.3.6.1-42}$$

Equation (12.3.6.1-42) is now transformed to the N frame using generalized Equation (3.4-6):

$$\begin{aligned}\delta\dot{\underline{R}}_H^N &= \delta\underline{v}_H^N + \underline{\varepsilon}^N \times \underline{v}^N - \underline{\omega}_{EN}^N \times \left(\delta R\,\underline{u}_{ZN}^N\right) - \underline{\omega}_{EN}^N \times \delta\underline{R}_H^N \\ &= \delta\underline{v}_H^N + \underline{\varepsilon}^N \times \underline{v}^N - \underline{\omega}_{EN}^N \times \left(\delta R\,\underline{u}_{ZN}^N + \delta\underline{R}_H^N\right)\end{aligned} \tag{12.3.6.1-43}$$

or with (12.3.6.1-16):

$$\delta\dot{\underline{R}}_H^N = \delta\underline{v}_H^N + \underline{\varepsilon}^N \times \underline{v}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N \qquad (12.3.6.1\text{-}44)$$

To complete the development, the equation for $\delta\dot{\underline{R}}^N$ is obtained from (12.3.6.1-44) by substituting the time derivative of Equation (12.3.6.1-16):

$$\delta\dot{\underline{R}}^N = \delta\dot{\underline{R}}_H^N + \delta\dot{R}\, \underline{u}_{ZN}^N \qquad (12.3.6.1\text{-}45)$$

The $\delta\dot{R}$ term in (12.3.6.1-45) is from (12.2.3-23) and $\delta\dot{h}$ in Equations (12.3.4-38):

$$\delta\dot{R} = \underline{u}_{ZN}^N \cdot \delta\underline{v}^N - C_3 \left(\delta R - \delta h_{Prsr}\right) \qquad (12.3.6.1\text{-}46)$$

Substituting (12.3.6.1-44) and (12.3.6.1-46) into (12.3.6.1-45) then yields:

$$\begin{aligned}
\delta\dot{\underline{R}}^N &= \delta\underline{v}_H^N + \underline{\varepsilon}^N \times \underline{v}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N + \left[\underline{u}_{ZN}^N \cdot \delta\underline{v}^N - C_3 \left(\delta R - \delta h_{Prsr}\right)\right] \underline{u}_{ZN}^N \\
&= \delta\underline{v}_H^N + \left(\underline{u}_{ZN}^N \cdot \delta\underline{v}^N\right) \underline{u}_{ZN}^N + \underline{\varepsilon}^N \times \underline{v}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^N
\end{aligned}$$

or, identifying $\left(\underline{u}_{ZN}^N \cdot \delta\underline{v}^N\right) \underline{u}_{ZN}^N$ as the vertical component of $\delta\underline{v}^N$:

$$\delta\dot{\underline{R}}^N = \delta\underline{v}^N + \underline{\varepsilon}^N \times \underline{v}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^N \qquad (12.3.6.1\text{-}47)$$

The final form of $\delta\dot{\underline{R}}^N$ is obtained by substituting (12.3.6.1-9) for $\delta\underline{v}^N + \underline{\varepsilon}^N \times \underline{v}^N$:

$$\delta\dot{\underline{R}}^N = \delta\underline{V}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^N \qquad (12.3.6.1\text{-}48)$$

Equation (12.3.6.1-48) is identical to the $\delta\dot{\underline{R}}^N$ expression in Equations (12.3.3-6).

12.3.6.2 N FRAME DEFINED ERROR PARAMETER RATE EQUATIONS FROM E FRAME DEFINED ERROR PARAMETER RATE EQUATIONS

In this section we derive the time rate equations for the $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh navigation error parameters beginning with the Section 12.3.3 differential equations for $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ (summarized in Equations (12.3.3-6)) and incorporating the Section 12.2.1 - 12.2.3

equivalencies between $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ and $\underline{\gamma}^N$, $\delta\underline{v}^N$, $\underline{\varepsilon}^N$, δh. The development cannot create the vertical component of $\dot{\underline{\varepsilon}}^N$ which must be specified independently.

The development for $\underline{\varepsilon}^N$ begins in the E Frame and then is transformed to the N Frame. We first formally specify $\underline{\varepsilon}^E$ as the sum of its horizontal and vertical components:

$$\underline{\varepsilon}^E = \underline{\varepsilon}_H^E + \varepsilon_{ZN}\, \underline{u}_{ZN}^E \tag{12.3.6.2-1}$$

where

H = Subscript designator for horizontal component of the associated vector quantity.

$\underline{\varepsilon}_H^E$ = Horizontal component of $\underline{\varepsilon}^E$ (i.e., perpendicular to the local vertical $\underline{u}_{ZN}^E$).

ε_{ZN} = Vertical component of $\underline{\varepsilon}^E$ (i.e., parallel to the local vertical $\underline{u}_{ZN}^E$).

From Equations (12.3.6.2-1) and (12.2.3-25) in the E Frame we can write:

$$\underline{\varepsilon}_H^E = \frac{1}{R}\left(\underline{u}_{ZN}^E \times \delta\underline{R}^E\right) \tag{12.3.6.2-2}$$

The derivative of (12.3.6.2-2) is:

$$\dot{\underline{\varepsilon}}_H^E = \frac{1}{R}\left(\underline{u}_{ZN}^E \times \delta\dot{\underline{R}}^E\right) - \frac{\dot{R}}{R^2}\left(\underline{u}_{ZN}^E \times \delta\underline{R}^E\right) + \frac{1}{R}\left(\dot{\underline{u}}_{ZN}^E \times \delta\underline{R}^E\right) \tag{12.3.6.2-3}$$

or with (12.3.6.1-25) for $\dot{\underline{u}}_{ZN}^E$ and (12.3.6.2-2):

$$\dot{\underline{\varepsilon}}_H^E = \frac{1}{R}\left(\underline{u}_{ZN}^E \times \delta\dot{\underline{R}}^E\right) - \frac{\dot{R}}{R}\,\underline{\varepsilon}_H^E + \frac{1}{R}\left(\underline{\omega}_{EN}^E \times \underline{u}_{ZN}^E\right) \times \delta\underline{R}^E \tag{12.3.6.2-4}$$

Using generalized vector rate transformation Equation (3.4-6), the $\delta\dot{\underline{R}}^E$ term in (12.3.6.2-4) can be replaced by the E Frame version of the Equations (12.3.3-6) $\delta\dot{\underline{R}}^N$ expression with (12.2.3-23) for δR:

$$\delta\dot{\underline{R}}^E = \delta\underline{V}^E - C_3\left(\delta h - \delta h_{Prsr}\right)\underline{u}_{ZN}^E \tag{12.3.6.2-5}$$

Equivalency Equation (12.2.2-5) is in the E Frame:

$$\delta\underline{V}^E = \delta\underline{v}^E + \underline{\varepsilon}^E \times \underline{v}^E \tag{12.3.6.2-6}$$

With (12.3.6.2-5) and (12.3.6.2-6), the $\underline{u}_{ZN}^{E} \times \delta\dot{\underline{R}}^{E}$ term in (12.3.6.2-4) is:

$$\underline{u}_{ZN}^{E} \times \delta\dot{\underline{R}}^{E} = \underline{u}_{ZN}^{E} \times \delta\underline{v}^{E} + \underline{u}_{ZN}^{E} \times \left(\underline{\varepsilon}^{E} \times \underline{v}^{E}\right) \qquad (12.3.6.2\text{-}7)$$

The $\underline{u}_{ZN}^{E} \times \left(\underline{\varepsilon}^{E} \times \underline{v}^{E}\right)$ term in (12.3.6.2-7) can be expanded using the Equation (3.1.1-16) vector triple cross-product rule, Equation (12.3.6.1-34) and (12.3.6.2-1):

$$\underline{u}_{ZN}^{E} \times \left(\underline{\varepsilon}^{E} \times \underline{v}^{E}\right) = \underline{\varepsilon}^{E}\left(\underline{u}_{ZN}^{E} \cdot \underline{v}^{E}\right) - \underline{v}^{E}\left(\underline{u}_{ZN}^{E} \cdot \underline{\varepsilon}^{E}\right) = \dot{R}\,\underline{\varepsilon}^{E} - \varepsilon_{ZN}\,\underline{v}^{E} \qquad (12.3.6.2\text{-}8)$$

Substituting (12.3.6.2-7) with (12.3.6.2-8) into (12.3.6.2-4) and applying (12.3.6.2-1) then yields:

$$\begin{aligned} \dot{\underline{\varepsilon}}_{H}^{E} &= \frac{1}{R}\left(\underline{u}_{ZN}^{E} \times \delta\underline{v}^{E}\right) + \frac{\dot{R}}{R}\left(\underline{\varepsilon}^{E} - \underline{\varepsilon}_{H}^{E}\right) - \frac{\varepsilon_{ZN}}{R}\,\underline{v}^{E} + \frac{1}{R}\left(\underline{\omega}_{EN}^{E} \times \underline{u}_{ZN}^{E}\right) \times \delta\underline{R}^{E} \\ &= \frac{1}{R}\left(\underline{u}_{ZN}^{E} \times \delta\underline{v}^{E}\right) + \frac{\dot{R}}{R}\,\varepsilon_{ZN}\,\underline{u}_{ZN}^{E} - \frac{\varepsilon_{ZN}}{R}\,\underline{v}^{E} + \frac{1}{R}\left(\underline{\omega}_{EN}^{E} \times \underline{u}_{ZN}^{E}\right) \times \delta\underline{R}^{E} \end{aligned} \qquad (12.3.6.2\text{-}9)$$

With (12.3.6.1-34) and (12.3.6.1-30), the ε_{ZN} terms in (12.3.6.2-9) combine as follows:

$$\frac{\dot{R}}{R}\,\varepsilon_{ZN}\,\underline{u}_{ZN}^{E} - \frac{\varepsilon_{ZN}}{R}\,\underline{v}^{E} = \frac{\varepsilon_{ZN}}{R}\left(\dot{R}\,\underline{u}_{ZN}^{E} - \underline{v}^{E}\right) = -\,\varepsilon_{ZN}\,\frac{1}{R}\,\underline{v}_{H}^{E} = -\,\varepsilon_{ZN}\left(\underline{\omega}_{EN}^{E} \times \underline{u}_{ZN}^{E}\right) \qquad (12.3.6.2\text{-}10)$$

The $\frac{1}{R}\left(\underline{\omega}_{EN}^{E} \times \underline{u}_{ZN}^{E}\right) \times \delta\underline{R}^{E}$ term in (12.3.6.2-9) can be expanded by noting that the vertical component of $\underline{\omega}_{EN}^{E}$ has no contribution to the cross-product with $\underline{u}_{ZN}^{E}$, applying the Equation (3.1.1-16) vector triple cross-product rule, and using Equation (12.3.6.1-16) in the E Frame:

$$\begin{aligned} \frac{1}{R}\left(\underline{\omega}_{EN}^{E} \times \underline{u}_{ZN}^{E}\right) \times \delta\underline{R}^{E} &= \frac{1}{R}\left(\underline{\omega}_{EN_H}^{E} \times \underline{u}_{ZN}^{E}\right) \times \delta\underline{R}^{E} \\ &= \frac{1}{R}\,\underline{u}_{ZN}^{E}\left(\underline{\omega}_{EN_H}^{E} \cdot \delta\underline{R}^{E}\right) - \frac{1}{R}\,\underline{\omega}_{EN_H}^{E}\left(\underline{u}_{ZN}^{E} \cdot \delta\underline{R}^{E}\right) \\ &= \frac{1}{R}\left(\underline{\omega}_{EN_H}^{E} \cdot \delta\underline{R}_{H}^{E}\right)\underline{u}_{ZN}^{E} - \frac{\delta R}{R}\,\underline{\omega}_{EN_H}^{E} \end{aligned} \qquad (12.3.6.2\text{-}11)$$

where

$\underline{\omega}_{EN_H}^{E}$ = Horizontal component of $\underline{\omega}_{EN}^{E}$.

Substituting Equations (12.3.6.2-10) and (12.3.6.2-11) into (12.3.6.2-9) and applying (12.3.6.1-27) then gives:

$$\begin{aligned}\dot{\underline{\varepsilon}}_H^E &= \frac{1}{R}\left(\underline{u}_{ZN}^E \times \delta\underline{v}^E\right) - \frac{\delta R}{R}\,\underline{\omega}_{EN_H}^E - \varepsilon_{ZN}\left(\underline{\omega}_{EN}^E \times \underline{u}_{ZN}^E\right) + \frac{1}{R}\left(\underline{\omega}_{EN_H}^E \cdot \delta\underline{R}_H^E\right)\underline{u}_{ZN}^E \\ &= \frac{1}{R}\left(\underline{u}_{ZN}^E \times \delta\underline{v}^E\right) - \frac{\delta R}{R^2}\left(\underline{u}_{ZN}^E \times \underline{v}^E\right) - \varepsilon_{ZN}\left(\underline{\omega}_{EN}^E \times \underline{u}_{ZN}^E\right) + \frac{1}{R}\left(\underline{\omega}_{EN_H}^E \cdot \delta\underline{R}_H^E\right)\underline{u}_{ZN}^E\end{aligned} \qquad (12.3.6.2\text{-}12)$$

or with the horizontal component of Equation (12.3.6.1-38) for the two terms directly on the right of the equal sign, and $\delta\rho_{ZN}$ recognized as the vertical component of $\delta\underline{\omega}_{EN}$:

$$\dot{\underline{\varepsilon}}_H^E = \delta\underline{\omega}_{EN_H}^E - \varepsilon_{ZN}\left(\underline{\omega}_{EN}^E \times \underline{u}_{ZN}^E\right) + \frac{1}{R}\left(\underline{\omega}_{EN_H}^E \cdot \delta\underline{R}_H^E\right)\underline{u}_{ZN}^E \qquad (12.3.6.2\text{-}13)$$

where

$\delta\underline{\omega}_{EN_H}^E$ = Horizontal component of $\delta\underline{\omega}_{EN}^E$.

Equation (12.3.6.2-13) is now transformed to the N Frame using generalized Equation (3.4-6), recognizing the angular rate of the E relative to the N Frame as the negative of the angular rate of N relative to E, and substituting (12.3.6.2-1) in the N Frame:

$$\begin{aligned}\dot{\underline{\varepsilon}}_H^N &= \delta\underline{\omega}_{EN_H}^N - \varepsilon_{ZN}\left(\underline{\omega}_{EN}^N \times \underline{u}_{ZN}^N\right) + \frac{1}{R}\left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right)\underline{u}_{ZN}^N + \underline{\omega}_{NE}^N \times \underline{\varepsilon}_H^N \\ &= \delta\underline{\omega}_{EN_H}^N - \underline{\omega}_{EN}^N \times \left(\underline{\varepsilon}_H^N + \varepsilon_{ZN}\,\underline{u}_{ZN}^N\right) + \frac{1}{R}\left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right)\underline{u}_{ZN}^N \\ &= -\,\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \frac{1}{R}\left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right)\underline{u}_{ZN}^N + \delta\underline{\omega}_{EN_H}^N\end{aligned} \qquad (12.3.6.2\text{-}14)$$

Equation (12.3.6.2-14) can be simplified by identifying the $\frac{1}{R}\left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right)\underline{u}_{ZN}^N$ term as the vertical component of $\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N$. This can be demonstrated by first expanding $\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N$ as the sum of its horizontal and vertical components using general vector cross-product equivalencies. Consider the cross-product of two arbitrary vectors $\underline{V}_1$ and $\underline{V}_2$ with components along and perpendicular to an arbitrary unit vector $\underline{u}$:

$$\underline{V}_1 = \underline{V}_{1_\perp} + \left(\underline{u}^N \cdot \underline{V}_1\right)\underline{u}^N \qquad \underline{V}_2 = \underline{V}_{2_\perp} + \left(\underline{u}^N \cdot \underline{V}_2\right)\underline{u}^N \qquad (12.3.6.2\text{-}15)$$

where

$\underline{u}$ = Arbitrary unit vector.

$\underline{V}_{1_\perp}, \underline{V}_{2_\perp}$ = Components of $\underline{V}_1$, $\underline{V}_2$ perpendicular to $\underline{u}$.

With (12.3.6.2-15), the cross-product of $\underline{V}_1$ with $\underline{V}_2$ is:

$$\begin{aligned}\underline{V}_1 \times \underline{V}_2 &= \underline{V}_{1_\perp} \times \underline{V}_{2_\perp} + \left(\underline{u}^N \cdot \underline{V}_1\right)\left(\underline{u}^N \times \underline{V}_{2_\perp}\right) - \left(\underline{u}^N \cdot \underline{V}_2\right)\left(\underline{u}^N \times \underline{V}_{1_\perp}\right) \\ &= \underline{V}_{1_\perp} \times \underline{V}_{2_\perp} + \left(\underline{u}^N \cdot \underline{V}_1\right)\left(\underline{u}^N \times \underline{V}_2\right) - \left(\underline{u}^N \cdot \underline{V}_2\right)\left(\underline{u}^N \times \underline{V}_1\right)\end{aligned} \tag{12.3.6.2-16}$$

From the definition of the cross-product, the $\left(\underline{u}^N \times \underline{V}_1\right)$ and $\left(\underline{u}^N \times \underline{V}_2\right)$ terms in (12.3.6.2-16) are perpendicular to $\underline{u}$. From their definition above, each of the $\underline{V}_{1_\perp}$, $\underline{V}_{2_\perp}$ terms lie in a plane perpendicular to $\underline{u}$, hence, their cross-product in (12.3.6.2-16) lies along $\underline{u}$. We conclude that the $\underline{V}_{1_\perp} \times \underline{V}_{2_\perp}$ term in (12.3.6.2-16) represents the component of $\underline{V}_1 \times \underline{V}_2$ along $\underline{u}$ while the remaining terms represent the $\underline{V}_1 \times \underline{V}_2$ component perpendicular to $\underline{u}$. Therefore:

$$\underline{V}_1 \times \underline{V}_2 = \left(\underline{V}_1 \times \underline{V}_2\right)_\perp + \underline{V}_{1_\perp} \times \underline{V}_{2_\perp} \tag{12.3.6.2-17}$$

where

$\left(\underline{V}_1 \times \underline{V}_2\right)_\perp$ = Component of $\underline{V}_1 \times \underline{V}_2$ perpendicular to $\underline{u}$.

We now apply general Equation (12.3.6.2-17) to $\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N$ by identifying $\underline{u}$ as the unit vector along the local vertical vector $\underline{u}_{ZN}^N$ and the $\perp$ subscript as referring to the horizontal plane H (i.e., perpendicular to the local vertical $\underline{u}_{ZN}^N$):

$$\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N = \left(\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N\right)_H + \underline{\omega}_{EN_H}^N \times \underline{\varepsilon}_H^N \tag{12.3.6.2-18}$$

The $\underline{\omega}_{EN_H}^N \times \underline{\varepsilon}_H^N$ term in (12.3.6.2-18) can be expanded using Equation (12.3.6.2-2) in the N Frame and the Equation (3.1.1-16) vector triple cross-product identity:

$$\underline{\omega}_{EN_H}^N \times \underline{\varepsilon}_H^N = \underline{\omega}_{EN_H}^N \times \frac{1}{R}\left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) = \frac{1}{R}\left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right)\underline{u}_{ZN}^N \tag{12.3.6.2-19}$$

which is the term of interest in Equation (12.3.6.2-14), and which from Equations (12.3.6.2-18) and (12.3.6.2-19), represents the component of $\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N$ along the local vertical $\underline{u}_{ZN}^N$ as stipulated.

With (12.3.6.2-18) and (12.3.6.2-19), Equation (12.3.6.2-14) assumes the final form:

$$\dot{\underline{\varepsilon}}_H^N = -\left(\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N\right)_H + \delta\underline{\omega}_{EN_H}^N \qquad (12.3.6.2\text{-}20)$$

Equation (12.3.6.2-20) is the horizontal component of the $\dot{\underline{\varepsilon}}^N$ expression in Equations (12.3.4-39).

The vertical component of $\dot{\underline{\varepsilon}}^N$ (i.e., $\dot{\varepsilon}_{ZN}$) cannot be derived from the $\underline{\psi}^N$, $\delta\underline{V}^N$, $\delta\underline{R}^N$ rate equations and must be developed independently. Also note that the complete $\underline{\varepsilon}^N$ vector (including the vertical component) is required in the (12.3.6.2-20) $\dot{\underline{\varepsilon}}_H^N$ equation. Equation (12.3.5-21) provides the following version of $\dot{\varepsilon}_{ZN}$:

$$\dot{\varepsilon}_{ZN} = -\frac{1}{R}\left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right) + \delta\rho_{ZN} \qquad (12.3.6.2\text{-}21)$$

The $\delta\underline{R}_H^N$ term in (12.3.6.2-21) is from Equation (12.3.6.1-16):

$$\delta\underline{R}_H^N = R\left(\underline{\varepsilon}^N \times \underline{u}_{ZN}^N\right) = R\left(\underline{\varepsilon}_H^N \times \underline{u}_{ZN}^N\right) \qquad (12.3.6.2\text{-}22)$$

hence, (12.3.6.2-21) is equivalently:

$$\dot{\varepsilon}_{ZN} = -\underline{\omega}_{EN_H}^N \cdot \left(\underline{\varepsilon}_H^N \times \underline{u}_{ZN}^N\right) + \delta\rho_{ZN} \qquad (12.3.6.2\text{-}23)$$

If Section 12.3.5 is reviewed, it will be remembered that Equation (12.3.6.2-21) is derived from the full $\dot{\underline{\varepsilon}}^N$ expression in Equations (12.3.4-39) which in turn was developed from basic definitions. Thus, the combination of Equations (12.3.6.2-20) and (12.3.6.2-23) is the Equations (12.3.4-39) expression:

$$\dot{\underline{\varepsilon}}^N = -\underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \qquad (12.3.6.2\text{-}24)$$

The equation for altitude error rate $\delta\dot{h}$ is derived from the $\delta\dot{\underline{R}}^N$ equation by first writing the derivative of equivalency Equation (12.2.3-23):

$$\delta\dot{h} = \delta\dot{R} \qquad (12.3.6.2\text{-}25)$$

The $\delta\dot{R}$ term in (12.3.6.2-25) is the derivative of the δR expression in (12.3.3-6):

$$\delta\dot{R} = \underline{u}_{ZN}^N \cdot \delta\dot{\underline{R}}^N \qquad (12.3.6.2\text{-}26)$$

Combining (12.3.6.2-25), (12.3.6.2-26), the Equations (12.3.3-6) expression for $\delta \dot{\underline{R}}^N$, equivalency Equation (12.2.3-23) for δR, and equivalency Equation (12.2.2-5) for $\delta \underline{V}^N$ gives:

$$\delta \dot{h} = \underline{u}_{ZN}^N \cdot \delta \underline{v}^N - C_3 \left(\delta h - \delta h_{Prsr}\right) + \underline{u}_{ZN}^N \cdot \left(\underline{\varepsilon}^N \times \underline{v}^N\right) - \underline{u}_{ZN}^N \cdot \left(\underline{\omega}_{EN}^N \times \delta \underline{R}^N\right) \qquad (12.3.6.2\text{-}27)$$

The last two terms in (12.3.6.2-27) are now shown to equal zero. Beginning with the $\underline{u}_{ZN}^N \cdot \left(\underline{\varepsilon}^N \times \underline{v}^N\right)$ terms, we apply generalized Equation (12.3.6.2-17) to show that:

$$\underline{\varepsilon}^N \times \underline{v}^N = \left(\underline{\varepsilon}^N \times \underline{v}^N\right)_H + \underline{\varepsilon}_H^N \times \underline{v}_H^N \qquad (12.3.6.2\text{-}28)$$

where

H = Designator for horizontal component (i.e., perpendicular to local vertical vector $\underline{u}_{ZN}^N$).

With (12.3.6.2-28), we see then that:

$$\underline{u}_{ZN}^N \cdot \left(\underline{\varepsilon}^N \times \underline{v}^N\right) = \underline{u}_{ZN}^N \cdot \left(\underline{\varepsilon}_H^N \times \underline{v}_H^N\right) \qquad (12.3.6.2\text{-}29)$$

The $\underline{u}_{ZN}^N \cdot \left(\underline{\omega}_{EN}^N \times \delta \underline{R}^N\right)$ term in (12.3.6.2-27) is expanded by first applying Equation (12.3.6.2-17) to $\left(\underline{\omega}_{EN}^N \times \delta \underline{R}^N\right)$:

$$\left(\underline{\omega}_{EN}^N \times \delta \underline{R}^N\right) = \left(\underline{\omega}_{EN}^N \times \delta \underline{R}^N\right)_H + \underline{\omega}_{EN_H}^N \times \delta \underline{R}_H^N \qquad (12.3.6.2\text{-}30)$$

Taking the dot product of (12.3.6.2-30) with $\underline{u}_{ZN}^N$ cancels the first term on the right of the equal sign so that, with (12.3.6.2-22) for $\delta \underline{R}_H^N$ and triple vector cross-product identity (3.1.1-16), we obtain:

$$\begin{aligned} \underline{u}_{ZN}^N \cdot \left(\underline{\omega}_{EN}^N \times \delta \underline{R}^N\right) &= R\, \underline{u}_{ZN}^N \cdot \left[\underline{\omega}_{EN_H}^N \times \left(\underline{\varepsilon}_H^N \times \underline{u}_{ZN}^N\right)\right] \\ &= - R\, \underline{u}_{ZN}^N \cdot \left[\underline{u}_{ZN}^N \left(\underline{\omega}_{EN_H}^N \cdot \underline{\varepsilon}_H^N\right)\right] = - R\, \underline{\omega}_{EN_H}^N \cdot \underline{\varepsilon}_H^N \end{aligned} \qquad (12.3.6.2\text{-}31)$$

The $\underline{\omega}_{EN_H}^N$ term in (12.3.6.2-31) is from Equation (12.3.6.1-27) in the N Frame:

$$\underline{\omega}_{EN_H}^N = \frac{1}{R} \left(\underline{u}_{ZN}^N \times \underline{v}^N\right) = - \frac{1}{R} \left(\underline{v}_H^N \times \underline{u}_{ZN}^N\right) \qquad (12.3.6.2\text{-}32)$$

Substituting (12.3.6.2-32) into (12.3.6.2-31) and applying the Equation (3.1.1-35) mixed vector dot/cross-product identity yields:

$$\underline{u}_{ZN}^{N} \cdot \left(\underline{\omega}_{EN}^{N} \times \delta\underline{R}^{N}\right) = \left(\underline{v}_{H}^{N} \times \underline{u}_{ZN}^{N}\right) \cdot \underline{\varepsilon}_{H}^{N} = \underline{u}_{ZN}^{N} \cdot \left(\underline{\varepsilon}_{H}^{N} \times \underline{v}_{H}^{N}\right) \qquad (12.3.6.2\text{-}33)$$

Comparing (12.3.6.2-33) with (12.3.6.2-29), we see then as stipulated that:

$$\underline{u}_{ZN}^{N} \cdot \left(\underline{\varepsilon}^{N} \times \underline{v}^{N}\right) - \underline{u}_{ZN}^{N} \cdot \left(\underline{\omega}_{EN}^{N} \times \delta\underline{R}^{N}\right) = 0 \qquad (12.3.6.2\text{-}34)$$

Using (12.3.6.2-34), Equation (12.3.6.2-27) simplifies to :

$$\delta\dot{h} = \underline{u}_{ZN}^{N} \cdot \delta\underline{v}^{N} - C_3 \left(\delta h - \delta h_{Prsr}\right) \qquad (12.3.6.2\text{-}35)$$

which is identical to the $\delta\dot{h}$ expression in Equations (12.3.4-39).

The $\delta\dot{\underline{V}}^{N}$ equation is derived from $\delta\dot{\underline{v}}^{N}$ starting with the Equation (12.2.2-5) equivalency:

$$\delta\underline{V}^{N} = \delta\underline{v}^{N} + \underline{\varepsilon}^{N} \times \underline{v}^{N} \qquad (12.3.6.2\text{-}36)$$

The derivative of (12.3.6.2-36) when rearranged gives:

$$\delta\dot{\underline{v}}^{N} = \delta\dot{\underline{V}}^{N} - \dot{\underline{\varepsilon}}^{N} \times \underline{v}^{N} - \underline{\varepsilon}^{N} \times \dot{\underline{v}}^{N} \qquad (12.3.6.2\text{-}37)$$

The $\dot{\underline{v}}^{N}$ term in (12.3.6.2-37) is from Equations (12.1.2-6):

$$\dot{\underline{v}}^{N} = C_B^N \, \underline{a}_{SF}^{B} + \underline{g}_{P}^{N} - \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \underline{v}^{N} - e_{vc_1} \underline{u}_{ZN}^{N} \qquad (12.3.6.2\text{-}38)$$

Using the Equation (12.3.6.1-18) approximation for $\underline{g}_{P}^{N}$ in Equation (12.3.6.2-38), the $\underline{\varepsilon}^{N} \times \dot{\underline{v}}^{N}$ term in (12.3.6.2-37) becomes:

$$\underline{\varepsilon}^{N} \times \dot{\underline{v}}^{N} = \underline{\varepsilon}^{N} \times \left(\underline{a}_{SF}^{N} - g\, \underline{u}_{ZN}^{N} - \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \underline{v}^{N} - e_{vc_1} \underline{u}_{ZN}^{N}\right) \qquad (12.3.6.2\text{-}39)$$

The $\delta\dot{\underline{V}}^{N}$ term in (12.3.6.2-37) is from Equations (12.3.3-6) using (12.2.3-23) for δR:

$$\begin{aligned}\delta\dot{\underline{V}}^{N} = {} & C_B^N \, \delta\underline{a}_{SF}^{B} + \underline{a}_{SF}^{N} \times \underline{\psi}^{N} - \frac{g}{R}\, \delta\underline{R}_{H}^{N} - \left(\underline{\omega}_{IE}^{N} + \underline{\omega}_{IN}^{N}\right) \times \delta\underline{V}^{N} + \delta\underline{g}_{Mdl}^{N} \\ & + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta h + C_2\, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^{N} - e_{vc_1} \frac{1}{R}\, \delta\underline{R}_{H}^{N}\end{aligned} \qquad (12.3.6.2\text{-}40)$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

with $\underline{\omega}_{IN}^{N}$ from Equation (12.3.6.1-4), $\delta\underline{R}_{H}^{N}$ from (12.3.6.2-22), $\delta\underline{V}^{N}$ from (12.3.6.2-36), and the Equation (12.2.1-17) equivalency for $\underline{\psi}^{N}$:

$$\underline{\psi}^{N} = \underline{\gamma}^{N} - \underline{\varepsilon}^{N} \tag{12.3.6.2-41}$$

Note in Equation (12.3.6.2-40) that we have included the $e_{vc_1} \frac{1}{R} \delta\underline{R}_{H}^{N}$ term which was dropped in the development of $\delta\dot{\underline{V}}^{N}$ from its E Frame equivalent as negligible (See discussion surrounding Equation (12.3.2-34)).

An equation for $\dot{\underline{\varepsilon}}^{N}$ in (12.3.6.2-37) has already been derived in this section as Equation (12.3.6.2-24).

We now substitute (12.3.6.2-24), (12.3.6.2-39) and (12.3.6.2-40) with (12.3.6.2-41), (12.3.6.1-4), (12.3.6.2-22) and (12.3.6.2-36) into (12.3.6.2-37) to obtain for $\delta\dot{\underline{v}}^{N}$:

$$\begin{aligned}
\delta\dot{\underline{v}}^{N} = \; & C_{B}^{N}\, \delta\underline{a}_{SF}^{B} + \underline{a}_{SF}^{N} \times \left(\underline{\gamma}^{N} - \underline{\varepsilon}^{N}\right) - g \left(\underline{\varepsilon}^{N} \times \underline{u}_{ZN}^{N}\right) \\
& - \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \left(\delta\underline{v}^{N} + \underline{\varepsilon}^{N} \times \underline{v}^{N}\right) + \delta\underline{g}_{Mdl}^{N} \\
& + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta h + C_2\, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^{N} - e_{vc_1} \left(\underline{\varepsilon}^{N} \times \underline{u}_{ZN}^{N}\right) \\
& - \left(- \underline{\omega}_{EN}^{N} \times \underline{\varepsilon}^{N} + \delta\underline{\omega}_{EN}^{N}\right) \times \underline{v}^{N} + \underline{a}_{SF}^{N} \times \underline{\varepsilon}^{N} \\
& + \underline{\varepsilon}^{N} \times \left[g\, \underline{u}_{ZN}^{N} + \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \underline{v}^{N} + e_{vc_1}\, \underline{u}_{ZN}^{N}\right]
\end{aligned} \tag{12.3.6.2-42}$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

or after rearrangement and cancellation of corresponding terms of opposite sign:

$$\begin{aligned}
\delta\dot{\underline{v}}^{N} = \; & C_{B}^{N}\, \delta\underline{a}_{SF}^{B} + \underline{a}_{SF}^{N} \times \underline{\gamma}^{N} - \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \delta\underline{v}^{N} + \underline{v}^{N} \times \delta\underline{\omega}_{EN}^{N} \\
& + \delta\underline{g}_{Mdl}^{N} + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta h + C_2\, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^{N} \\
& + \left(\underline{\omega}_{EN}^{N} \times \underline{\varepsilon}^{N}\right) \times \underline{v}^{N} + \underline{\varepsilon}^{N} \times \left[\left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \underline{v}^{N}\right] - \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \left(\underline{\varepsilon}^{N} \times \underline{v}^{N}\right)
\end{aligned} \tag{12.3.6.2-43}$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

The terms on the last line of Equation (12.3.6.2-43) can be rearranged as follows:

$$\begin{aligned}&\left(\underline{\omega}_{EN}^{N} \times \underline{\varepsilon}^{N}\right) \times \underline{v}^{N} + \underline{\varepsilon}^{N} \times \left[\left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \underline{v}^{N}\right] - \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \left(\underline{\varepsilon}^{N} \times \underline{v}^{N}\right) \\ &\quad = \left[\left(\underline{\omega}_{EN}^{N} \times \underline{\varepsilon}^{N}\right) \times \underline{v}^{N} + \underline{\varepsilon}^{N} \times \left(\underline{\omega}_{EN}^{N} \times \underline{v}^{N}\right) - \underline{\omega}_{EN}^{N} \times \left(\underline{\varepsilon}^{N} \times \underline{v}^{N}\right)\right] \\ &\quad + 2\left[\underline{\varepsilon}^{N} \times \left(\underline{\omega}_{IE}^{N} \times \underline{v}^{N}\right) - \underline{\omega}_{IE}^{N} \times \left(\underline{\varepsilon}^{N} \times \underline{v}^{N}\right)\right]\end{aligned} \tag{12.3.6.2-44}$$

The first square bracketed term in Equation (12.3.6.2-44) can be shown to be identically zero by rearrangement and application of generalized Equations (3.1.1-15) and (3.1.1-21):

$$\begin{aligned}&\left[\left(\underline{\omega}_{EN}^{N} \times \underline{\varepsilon}^{N}\right) \times \underline{v}^{N} + \underline{\varepsilon}^{N} \times \left(\underline{\omega}_{EN}^{N} \times \underline{v}^{N}\right) - \underline{\omega}_{EN}^{N} \times \left(\underline{\varepsilon}^{N} \times \underline{v}^{N}\right)\right] \\ &= \left[\left(\underline{\omega}_{EN}^{N} \times \underline{\varepsilon}^{N}\right) \times\right] \underline{v}^{N} - \left[\left(\underline{\omega}_{EN}^{N} \times\right)\left(\underline{\varepsilon}^{N} \times\right) \underline{v}^{N} - \left(\underline{\varepsilon}^{N} \times\right)\left(\underline{\omega}_{EN}^{N} \times\right) \underline{v}^{N}\right] = 0\end{aligned} \tag{12.3.6.2-45}$$

The second square bracketed term in (12.3.6.2-44) can be compressed by application of generalized Equations (3.1.1-15) and (3.1.1-21):

$$\begin{aligned}&2\left[\underline{\varepsilon}^{N} \times \left(\underline{\omega}_{IE}^{N} \times \underline{v}^{N}\right) - \underline{\omega}_{IE}^{N} \times \left(\underline{\varepsilon}^{N} \times \underline{v}^{N}\right)\right] = 2\left[\left(\underline{\varepsilon}^{N} \times\right)\left(\underline{\omega}_{IE}^{N} \times\right) \underline{v}^{N} - \left(\underline{\omega}_{IE}^{N} \times\right)\left(\underline{\varepsilon}^{N} \times\right) \underline{v}^{N}\right] \\ &\quad = 2\left(\underline{\varepsilon}^{N} \times \underline{\omega}_{IE}^{N}\right) \times \underline{v}^{N} = -2\left(\underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N}\right) \times \underline{v}^{N} = 2\, \underline{v}^{N} \times \left(\underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N}\right)\end{aligned} \tag{12.3.6.2-46}$$

Finally, we substitute (12.3.6.2-45) and (12.3.6.2-46) in (12.3.6.2-44), and the result into (12.3.6.2-43) to yield the desired expression for $\delta\dot{\underline{v}}^{N}$:

$$\begin{aligned}\delta\dot{\underline{v}}^{N} &= C_{B}^{N}\, \delta\underline{a}_{SF}^{B} + \underline{a}_{SF}^{N} \times \underline{\gamma}^{N} + \underline{v}^{N} \times \left[\delta\underline{\omega}_{EN}^{N} + 2\left(\underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N}\right)\right] \\ &- \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \delta\underline{v}^{N} + \delta\underline{g}_{Mdl}^{N} + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta h + C_2\, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^{N}\end{aligned} \tag{12.3.6.2-47}$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

which is identical to the $\delta\dot{\underline{v}}^{N}$ expression in Equations (12.3.4-39).

Development of the $\dot{\underline{\psi}}^{N}$ equation from $\dot{\underline{\gamma}}^{N}$ begins with the derivative of the rearranged Equation (12.2.1-17) equivalency:

$$\dot{\underline{\gamma}}^{N} = \dot{\underline{\psi}}^{N} + \dot{\underline{\varepsilon}}^{N} \tag{12.3.6.2-48}$$

The $\dot{\underline{\psi}}^{N}$ term in (12.3.6.2-48) is from Equations (12.3.3-6):

$$\dot{\underline{\psi}}^{N} = -\, C_{B}^{N}\, \delta\underline{\omega}_{IB}^{B} - \underline{\omega}_{IN}^{N} \times \underline{\psi}^{N} \tag{12.3.6.2-49}$$

with $\underline{\psi}^N$ from Equation (12.3.6.2-41). The $\dot{\underline{\varepsilon}}^N$ term in (12.3.6.2-48) equals the horizontal component derived previously in this section plus a specified vertical component as given in total by Equation (12.3.6.2-24).

Substituting (12.3.6.2-24) and (12.3.6.2-49) with (12.3.6.1-4) and (12.3.6.2-41) into (12.3.6.2-48) then yields:

$$\begin{aligned}\dot{\underline{\gamma}}^N &= - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \left(\underline{\gamma}^N - \underline{\varepsilon}^N\right) - \underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \\ &= - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \left(\underline{\omega}_{IN}^N - \underline{\omega}_{EN}^N\right) \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N\end{aligned} \qquad (12.3.6.2\text{-}50)$$

Application of Equation (12.3.6.1-4) for $\left(\underline{\omega}_{IN}^N - \underline{\omega}_{EN}^N\right)$ in (12.3.6.2-50) then obtains the desired final form for $\dot{\underline{\gamma}}^N$:

$$\dot{\underline{\gamma}}^N = - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \qquad (12.3.6.2\text{-}51)$$

which is identical to the $\dot{\underline{\gamma}}^N$ expression in Equations (12.3.4-39).

12.3.7 I FRAME DEFINED ERROR PARAMETER RATE EQUATIONS

At times it is convenient to define a set of navigation error rate equations for which the error parameters are defined relative to non-rotating inertial I Frame axes. This section develops such an equation set in the I Frame in which the errors are defined, and then transforms these equations to the N Frame.

12.3.7.1 I FRAME DEFINED ERROR PARAMETER RATE EQUATIONS IN THE I FRAME

We now direct our attention at the development of differential equations for the inertial I Frame defined error parameters $\underline{\varphi}$, $\delta\underline{\upsilon}$ and $\delta\underline{r}$ which have been defined by Equations (12.2.1-18), (12.2.2-9) and (12.2.3-27):

$$\left(\underline{\varphi}^I \times\right) \equiv I - \hat{C}_B^I \left(C_B^I\right)^T \qquad \delta\underline{\upsilon}^I \equiv \hat{\underline{\upsilon}}^I - \underline{\upsilon}^I \qquad \delta\underline{r}^I \equiv \hat{\underline{R}}^I - \underline{R}^I \qquad (12.3.7.1\text{-}1)$$

Equations (12.2.1-24) and (12.2.3-31) showed that $\underline{\varphi}$ and $\delta\underline{r}$ are identical to the $\underline{\psi}$ and $\delta\underline{R}$ error parameters defined in the E Frame by Equations (12.2.1-4) and (12.2.3-1). Thus, Equations (12.3.7.1-1) for I frame defined error effects can also be written as:

$$\left(\underline{\psi}^I \times\right) \equiv I - \hat{C}_B^I \left(C_B^I\right)^T \qquad \delta\underline{\upsilon}^I \equiv \hat{\underline{\upsilon}}^I - \underline{\upsilon}^I \qquad \delta\underline{R}^I \equiv \hat{\underline{R}}^I - \underline{R}^I \tag{12.3.7.1-2}$$

The rate equation for $\underline{\psi}^I$ is easily obtained by multiplication of the $\dot{\underline{\psi}}^E$ expression in Equations (12.3.2-43) by C_E^I and applying generalized vector rate transformation Equation (3.4-6):

$$C_E^I \, \dot{\underline{\psi}}^E = - C_B^I \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IE}^I \times \underline{\psi}^I \qquad \dot{\underline{\psi}}^I = C_E^I \, \dot{\underline{\psi}}^E + \underline{\omega}_{IE}^I \times \underline{\psi}^I \tag{12.3.7.1-3}$$

or in combination:

$$\dot{\underline{\psi}}^I = - C_B^I \, \delta\underline{\omega}_{IB}^B \tag{12.3.7.1-4}$$

The rate equation for $\delta\underline{R}^I$ can be obtained similarly from the $\delta\dot{\underline{R}}^E$ expression in Equations (12.3.2-43), application of Equation (12.2.2-15) for $\delta\underline{V}^I$, and use of generalized vector rate transformation Equation (3.4-6):

$$\begin{aligned}
&C_E^I \, \delta\dot{\underline{R}}^E = \delta\underline{V}^I - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^I \\
&\delta\underline{V}^I = \delta\underline{\upsilon}^I - \underline{\omega}_{IE}^I \times \delta\underline{R}^I \\
&\delta\dot{\underline{R}}^I = C_E^I \, \delta\dot{\underline{R}}^E + \underline{\omega}_{IE}^I \times \delta\underline{R}^I
\end{aligned} \tag{12.3.7.1-5}$$

or in combination:

$$\delta\dot{\underline{R}}^I = \delta\underline{\upsilon}^I - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^I \tag{12.3.7.1-6}$$

Development of the $\delta\underline{\upsilon}^I$ velocity error differential equation begins from the Equations (12.1.4-18) velocity rate expression repeated below:

$$\dot{\underline{\upsilon}}^I = C_B^I \, \underline{a}_{SF}^B + \underline{g}_P^I + \underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \underline{R}^I\right) - e_{vc_1} \, \underline{u}_{ZN}^I \tag{12.3.7.1-7}$$

The formal definition for the $\delta\underline{\upsilon}$ velocity error is given in the I Frame by Equations (12.3.7.1-2). Taking the analytical differential of Equation (12.3.7.1-7) and identifying the differential of $\underline{\upsilon}^I$ as $\delta\underline{\upsilon}^I$ from (12.3.7.1-2) yields:

$$\delta\dot{\underline{\upsilon}}^{I} = C_B^I\, \delta\underline{a}_{SF}^B + \delta C_B^I\, \underline{a}_{SF}^B + \delta\underline{g}_P^I + \underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \delta\underline{R}^I\right) - \delta e_{vc_1}\, \underline{u}_{ZN}^I - e_{vc_1}\, \delta\underline{u}_{ZN}^I \tag{12.3.7.1-8}$$

The δC_B^I term in (12.3.7.1-8) is obtained with Equation (12.2.1-3) as:

$$\begin{aligned} \delta C_B^I &= \delta\left(C_E^I\, C_B^E\right) = C_E^I\, \delta C_B^E = -\, C_E^I \left(\underline{\psi}^E \times\right) C_B^E \\ &= -\, C_E^I \left(\underline{\psi}^E \times\right) C_I^E\, C_B^I = -\, C_E^I \left(\underline{\psi}^E \times\right) C_E^{I\,T}\, C_B^I \end{aligned} \tag{12.3.7.1-9}$$

or with (3.1.1-39):

$$\delta C_B^I = -\left(\underline{\psi}^I \times\right) C_B^I \tag{12.3.7.1-10}$$

Equation (12.3.7.1-9) is based on the assumption that C_E^I is error free. The C_E^I matrix is a function of time in navigation and earth's inertial rotation rate magnitude. For this analysis we are assuming that the system clock timer is error free, hence because earth rate is a known constant (a very good approximation), C_E^I is error free. With modern day crystal oscillators being the inertial navigation timing source, the effect of timer error on navigation accuracy is generally small compared to the effect of inertial sensor errors.

The $\delta\underline{g}_P^I$ term in (12.3.7.1-8) is given by Equation (12.2.4-8):

$$\begin{aligned} &\delta\underline{g}_P^I \approx -\frac{g}{R}\, \delta\underline{R}_H^I + F(h)\, \frac{g}{R}\, \underline{u}_{ZN}^I\, \delta R + \delta\underline{g}_{Mdl}^I \\ &F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \end{aligned} \tag{12.3.7.1-11}$$

Following the same reasoning used to derive the Equation (12.2.4-1) $\delta\underline{g}_P^N$ expression (i.e., neglecting $\partial g_{P_{Up}}$ and $\partial g_{P_{North}}$), the $\underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \delta\underline{R}^I\right)$ term in (12.3.7.1-8) can be dropped as negligible compared to the δR term in $\delta\underline{g}_P^I$ Equation (12.3.7.1-11). However, it should be noted that the error is somewhat larger than neglecting the $\partial g_{P_{North}}$ term because $\partial g_{P_{North}}$ is based on the excellent approximation that horizontal plumb-bob gravity is zero at the earth's surface. Neglecting $\underline{\omega}_{IE}^I \times \left(\underline{\omega}_{IE}^I \times \delta\underline{R}^I\right)$ in Equation (12.3.7.1-8) ignores this higher accuracy assumption.

Following the same reasoning used to simplify Equation (12.3.2-21), the $e_{vc_1}\, \delta\underline{u}_{ZN}^I$ term in (12.3.7.1-8) can be dropped as negligible compared to the $\delta e_{vc_1}\, \underline{u}_{ZN}^I$ term. The remaining

$\delta e_{vc_1} \, \underline{u}_{ZN}^{I}$ term is given by Equation (12.3.2-33):

$$\delta e_{vc_1} = \delta e_{vc_3} + C_2 \left(\delta R - \delta h_{Prsr} \right) \tag{12.3.7.1-12}$$

Substituting (12.3.7.1-10) - (12.3.7.1-12) into (12.3.7.1-8) while neglecting $\underline{\omega}_{IE}^{I} \times \left(\underline{\omega}_{IE}^{I} \times \delta \underline{R}^{I} \right)$ and $e_{vc_1} \, \delta \underline{u}_{ZN}^{I}$ then obtains the I Frame form of the $\delta \underline{\upsilon}$ rate equation:

$$\begin{aligned} \delta \dot{\underline{\upsilon}}^{I} &= C_B^I \, \delta \underline{a}_{SF}^{B} + \underline{a}_{SF}^{I} \times \underline{\psi}^{I} - \frac{g}{R} \, \delta \underline{R}_H^I + \delta \underline{g}_{Mdl}^{I} \\ &\quad + \left[\left(F(h) \, \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^{I} \end{aligned} \tag{12.3.7.1-13}$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

The δe_{vc_3} term in (12.3.7.1-13) is obtained from Equations (12.3.2-35):

$$\delta \dot{e}_{vc_3} = C_1 \left(\delta R - \delta h_{Prsr} \right) \tag{12.3.7.1-14}$$

In summary, the I Frame version defined navigation error equations are given by Equations (12.3.7.1-4), (12.3.7.1-6), (12.3.7.1-13) and (12.3.7.1-14) with definitions for appropriate terms in Equations (12.3.2-43) transformed to the I frame:

$$\begin{aligned} \dot{\underline{\psi}}^{I} &= - C_B^I \, \delta \underline{\omega}_{IB}^{B} \\ \delta \dot{\underline{\upsilon}}^{I} &= C_B^I \, \delta \underline{a}_{SF}^{B} + \underline{a}_{SF}^{I} \times \underline{\psi}^{I} - \frac{g}{R} \, \delta \underline{R}_H^I + \delta \underline{g}_{Mdl}^{I} \\ &\quad + \left[\left(F(h) \, \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^{I} \\ F(h) &= 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \\ \delta \dot{\underline{R}}^{I} &= \delta \underline{\upsilon}^{I} - C_3 \left(\delta R - \delta h_{Prsr} \right) \underline{u}_{ZN}^{I} \\ \delta \dot{e}_{vc_3} &= C_1 \left(\delta R - \delta h_{Prsr} \right) \\ \delta \underline{R}_H^I &= \delta \underline{R}^{I} - \delta R \, \underline{u}_{ZN}^{I} \\ \delta R &= \underline{u}_{ZN}^{I} \cdot \delta \underline{R}^{I} \end{aligned} \tag{12.3.7.1-15}$$

12.3.7.2 I FRAME DEFINED ERROR PARAMETER RATE EQUATIONS IN THE N FRAME

The time rate differential equation in the N Frame for the $\underline{\psi}$ attitude error parameter is identical to $\dot{\underline{\psi}}^N$ in Equations (12.3.3-6). The time rate differential equations for the $\delta\underline{\upsilon}$, $\delta\underline{R}$ velocity/position error parameters in the N frame are derived by transforming $\delta\dot{\underline{\upsilon}}^I$, $\delta\dot{\underline{R}}^I$ in (12.3.7.1-15) to the N frame using generalized vector rate transformation Equation (3.4-6). We first find from (3.4-6) for $\delta\dot{\underline{\upsilon}}^I$:

$$\delta\dot{\underline{\upsilon}}^N = C_I^N \, \delta\dot{\underline{\upsilon}}^I + \underline{\omega}_{NI}^N \times \delta\underline{\upsilon}^N \tag{12.3.7.2-1}$$

where

$\underline{\omega}_{NI}^N$ = Relative angular rate of the I Frame with respect to the N Frame as projected on N Frame axes.

But we also know that:

$$\underline{\omega}_{NI}^N = - \underline{\omega}_{IN}^N \tag{12.3.7.2-2}$$

where

$\underline{\omega}_{IN}^N$ = Angular rate of the N Frame relative to the I Frame as projected on N Frame axes.

Thus, Equation (12.3.7.2-1) is equivalently:

$$\delta\dot{\underline{\upsilon}}^N = C_I^N \, \delta\dot{\underline{\upsilon}}^I - \underline{\omega}_{IN}^N \times \delta\underline{\upsilon}^N \tag{12.3.7.2-3}$$

With $\delta\dot{\underline{\upsilon}}^I$ from Equations (12.3.7.1-15) in (12.3.7.2-3) we then obtain for $\delta\dot{\underline{\upsilon}}^N$:

$$\begin{aligned} \delta\dot{\underline{\upsilon}}^N = {} & C_B^N \, \delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\psi}^N - \frac{g}{R} \, \delta\underline{R}_H^N - \underline{\omega}_{IN}^N \times \delta\underline{\upsilon}^N + \delta\underline{g}_{Mdl}^N \\ & + \left[\left(F(h) \, \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^N \end{aligned} \tag{12.3.7.2-4}$$

$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$

The $\delta\dot{\underline{R}}^N$ equation is obtained similarly from $\delta\dot{\underline{R}}^I$ in Equations (12.3.7.1-15):

$$\delta\dot{\underline{R}}^N = \delta\underline{\upsilon}^N - \underline{\omega}_{IN}^N \times \delta\underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr} \right) \underline{u}_{ZN}^N \tag{12.3.7.2-5}$$

With (12.3.7.2-4), (12.3.7.2-5) and appropriate elements from Equations (12.3.3-6), the $\underline{\psi}$, $\delta\underline{\upsilon}$, $\delta\underline{R}$ rate equations in the N Frame then are summarized as follows:

$$\dot{\underline{\psi}}^N = - C_B^N \, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\psi}^N$$

$$\delta\dot{\underline{\upsilon}}^N = C_B^N \, \delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\psi}^N - \frac{g}{R} \, \delta\underline{R}_H^N - \underline{\omega}_{IN}^N \times \delta\underline{\upsilon}^N + \delta\underline{g}_{Mdl}^N$$
$$+ \left[\left(F(h) \, \frac{g}{R} - C_2 \right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3} \right] \underline{u}_{ZN}^N$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0 \tag{12.3.7.2-6}$$

$$\delta\dot{\underline{R}}^N = \delta\underline{\upsilon}^N - \underline{\omega}_{IN}^N \times \delta\underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr} \right) \underline{u}_{ZN}^N$$

$$\delta\dot{e}_{vc_3} = C_1 \left(\delta R - \delta h_{Prsr} \right)$$

$$\delta\underline{R}_H^N = \delta\underline{R}^N - \delta R \, \underline{u}_{ZN}^N$$

$$\delta R = \underline{u}_{ZN}^N \cdot \delta\underline{R}^N$$

Note that Equations (12.3.7.2-6) in the N Frame reduce to (12.3.7.1-15) (in the I Frame) if we equate the N to the I Frame which sets $\underline{\omega}_{IN}^N$ to zero.

12.4 GENERAL STRAPDOWN INERTIAL SENSOR ERROR MODELS

The strapdown inertial navigation error equation sets derived in Section 12.3 (summarized in the N Frame by Equations (12.3.3-6), (12.3.4-39), (12.3.5-29) and (12.3.7.2-6)) contain the strapdown inertial angular rate sensor error term $\delta\underline{\omega}_{IB}^B$ and accelerometer error term $\delta\underline{a}_{SF}^B$. The $\delta\underline{\omega}_{IB}^B$ and $\delta\underline{a}_{SF}^B$ inertial sensor errors are defined as the difference between the input to the sensors (i.e., the true values that the sensor is measuring) and the compensated sensor output. The implication is that if the compensation is perfect, the compensated output will match the input, hence, there is no effective error that will impact navigation accuracy. In this section, we develop analytical models for the inertial sensor error terms based on the Section 8.1.1.1 and 8.1.1.2 sensor error characteristics.

We begin with the Section 8.1.1.1 description of the relationship between the angular rate sensor input and its uncompensated output as provided by Equation (8.1.1.1-2) and (8.1.1.1-3) with (8.1.1.1-5) - (8.1.1.1-7):

$$\underline{\omega}' = \Omega_{Wt}\ \underline{\omega}_{Puls} \tag{12.4-1}$$

$$\underline{\omega} = \underline{\omega}' - K_{Mis}\ \underline{\omega}' - \underline{K}_{Bias} - F_{Algn}^{-1}\ \partial\underline{\omega}_{Quant} - F_{Algn}^{-1}\ \delta\underline{\omega}_{Rand} \tag{12.4-2}$$

where

$\underline{\omega}$ = Strapdown angular rate sensor triad input vector.

$\underline{\omega}_{Puls}$ = Uncompensated angular rate sensor triad output pulse rate vector (pulses per second).

Ω_{Wt} = Angular rate sensor triad diagonal output pulse weighting matrix (radians per pulse).

F_{Algn} = Angular rate sensor triad alignment matrix.

$\underline{K}_{Bias}$ = Angular rate sensor effective bias vector.

K_{Mis} = Angular rate sensor effective misalignment matrix ($K_{Mis} = I - F_{Align}^{-1}$).

$\partial\underline{\omega}_{Quant}$ = Angular rate sensor triad pulse quantization error vector. Note that the quantization error is denoted here as $\partial\underline{\omega}_{Quant}$ while in Section 8.1.1.1 it is denoted as $\delta\underline{\omega}_{Quant}$. The distinction has been made here to differentiate errors that are partially correctable by compensation (the $\partial\underline{\omega}_{Quant}$ term) from residual errors remaining after compensation has been applied (the $\delta\underline{\omega}_{Quant}$ term to be introduced subsequently).

$\delta\underline{\omega}_{Rand}$ = Angular rate sensor triad random error vector.

A more precise set of definitions for the Equation(12.4-1) and (12.4-2) parameters is provided in Section 8.1.1.1.

The F_{Algn} matrix in (12.4-2) has, to first order, unity for its diagonal elements, and small misalignment terms for the off-diagonal elements. Hence, for error analysis purposes, Equation (12.4-2) can be approximated as:

$$\underline{\omega} \approx \underline{\omega}' - K_{Mis}\ \underline{\omega}' - \underline{K}_{Bias} - \partial\underline{\omega}_{Quant} - \delta\underline{\omega}_{Rand} \tag{12.4-3}$$

At this point we introduce nomenclature to differentiate between idealized sensor error compensation parameters and the compensation parameters actually utilized in the system software. We also introduce the $\sim$ notation to indicate outputs from sensors (raw or compensated) that contain errors. Thus, for the strapdown system software, the equivalent to Equations (12.4-1) and (12.4-3) is:

$$\underline{\tilde{\omega}}' = \Omega_{WtC}\,\underline{\omega}_{Puls} \tag{12.4-4}$$

$$\underline{\tilde{\omega}} \approx \underline{\tilde{\omega}}' - K_{MisC}\,\underline{\tilde{\omega}}' - \underline{K}_{BiasC} - \partial\underline{\omega}_{QuantC} \tag{12.4-5}$$

where

$\Omega_{WtC}, K_{MisC}, \underline{K}_{BiasC}, \partial\underline{\omega}_{QuantC}$ = Values for $\Omega_{Wt}, K_{Mis}, \underline{K}_{Bias}, \partial\underline{\omega}_{Quant}$ utilized for angular rate sensor compensation in the strapdown system computer.

$\tilde{}$ = Designator for inertial sensor outputs based on actual (imperfect) compensation applied in the strapdown system computer.

Equations (12.4-1) and (12.4-3) are considered to be the idealized version of Equations (12.4-4) and (12.4-5) where we now refine our definitions for the compensation terms:

$\Omega_{Wt}, K_{Mis}, \underline{K}_{Bias}, \partial\underline{\omega}_{Quant}$ = Idealized values of $\Omega_{WtC}, K_{MisC}, \underline{K}_{BiasC}, \partial\underline{\omega}_{QuantC}$ (i.e., the perfect compensation parameters).

Note that the primary difference between computer compensation Equations (12.4-4), (12.4-5) and the idealized (12.4-1), (12.4-3) equivalents, is the $\delta\underline{\omega}_{Rand}$ term which is inherently random by definition, hence, unaccountable in the actual system computer.

In developing the sensor error equation associated with Equation (12.4-5), we recognize that K_{MisC} is small, hence, for first order error analysis purposes, its $\underline{\tilde{\omega}}'$ term multiplier can be approximated by $\underline{\omega}'$ (the difference being $\delta\underline{\omega}'$ which, when multiplied by K_{MisC}, is second order). With this approximation, the error in $\underline{\tilde{\omega}}$ is now calculated as the difference between Equations (12.4-5) and (12.4-3) or:

$$\begin{aligned}\delta\underline{\omega} \equiv \underline{\tilde{\omega}} - \underline{\omega} = \underline{\tilde{\omega}}' - \underline{\omega}' + \left(K_{Mis} - K_{MisC}\right)\underline{\omega}' + \left(\underline{K}_{Bias} - \underline{K}_{BiasC}\right)\\ + \left(\partial\underline{\omega}_{Quant} - \partial\underline{\omega}_{QuantC}\right) + \delta\underline{\omega}_{Rand}\end{aligned} \tag{12.4-6}$$

where

$\delta\underline{\omega}$ = Residual error present in the compensated strapdown angular rate sensor triad output vector.

The $\underline{\tilde{\omega}}' - \underline{\omega}'$ term in (12.4-6) is the difference between Equations (12.4-4) and (12.4-1):

$$\underline{\tilde{\omega}}' - \underline{\omega}' = \left(\Omega_{WtC} - \Omega_{Wt}\right)\underline{\omega}_{Puls} \tag{12.4-7}$$

But from (12.4-1):

$$\underline{\omega}_{Puls} = \Omega_{Wt}^{-1}\, \underline{\omega}' \tag{12.4-8}$$

With (12.4-8), Equation (12.4-7) becomes:

$$\tilde{\underline{\omega}}' - \underline{\omega}' = \left[\Omega_{WtC}\, \Omega_{Wt}^{-1} - I\right] \underline{\omega}' \tag{12.4-9}$$

We introduce the following definitions for particular terms in (12.4-6) and (12.4-9):

$$\begin{aligned} &\delta K_{Scal} \equiv \Omega_{WtC}\, \Omega_{Wt}^{-1} - I \qquad && \delta K_{Mis} \equiv K_{Mis} - K_{MisC} \\ &\delta \underline{K}_{Bias} \equiv \underline{K}_{Bias} - \underline{K}_{BiasC} \qquad && \delta \underline{\omega}_{Quant} \equiv \partial \underline{\omega}_{Quant} - \partial \underline{\omega}_{QuantC} \end{aligned} \tag{12.4-10}$$

where

δK_{Mis}, $\delta \underline{K}_{Bias}$, $\delta \underline{\omega}_{Quant}$ = Residual errors in K_{Mis}, $\underline{K}_{Bias}$, $\partial \underline{\omega}_{Quant}$ remaining after applying compensation terms K_{MisC}, $\underline{K}_{BiasC}$, $\partial \underline{\omega}_{QuantC}$.

δK_{Scal} = Normalized residual scale factor error remaining in Ω_{Wt} after applying the $\Omega_{Wt\,C}$ scale factor matrix.

We also note that for error analysis purposes, $\underline{\omega}'$ in Equations (12.4-6) and (12.4-9) is, from Equation (12.4-3), approximately (to first order) equal to $\underline{\omega}$. Thus, with (12.4-9) and (12.4-10), Equation (12.4-6) to first order assumes the simpler form:

$$\delta \underline{\omega} = \delta K_{Scal} \underline{\omega} + \delta K_{Mis}\, \underline{\omega} + \delta \underline{K}_{Bias} + \delta \underline{\omega}_{Quant} + \delta \underline{\omega}_{Rand} \tag{12.4-11}$$

Combining the δK_{Scal}, δK_{Mis} terms in (12.4-11) and introducing the more definitive angular rate notation of previous sections obtains the final form:

$$\delta \underline{\omega}_{IB}^{B} = \delta K_{Scal/Mis}\, \underline{\omega}_{IB}^{B} + \delta \underline{K}_{Bias} + \delta \underline{\omega}_{Quant} + \delta \underline{\omega}_{Rand} \tag{12.4-12}$$

with

$$\delta K_{Scal/Mis} \equiv \delta K_{Scal} + \delta K_{Mis} \tag{12.4-13}$$

where

$\delta K_{Scal/Mis}$ = Effective residual scale-factor/misalignment error matrix remaining in Ω_{Wt} and K_{Mis} after applying compensation terms $\Omega_{Wt\,C}$ and K_{MisC}.

Equation (12.4-12) is the analytical model for $\delta \underline{\omega}_{IB}^{B}$ in N Frame navigation error Equations (12.3.3-6), (12.3.4-39), (12.3.5-29) and (12.3.7.2-6). The analytical models used for

$\delta K_{Scal/Mis}$ and $\delta \underline{K}_{Bias}$ in (12.4-12) depend on the angular rate sensor selected, its particular error characteristics and the associated compensation terms. The $\delta \underline{\omega}_{Rand}$ term is typically modeled as white noise. The $\delta \underline{\omega}_{Quant}$ term is typically represented as the time derivative of white noise which requires some revision in the navigation error equations (discussed in Section 12.5) for compatibility with standard random process equation formats (e.g., error state dynamic equation format discussed in Section 15.1 in which all noise terms are represented as white noise).

The steps that led to the Equation (12.4-12) $\delta \underline{\omega}_{IB}^{B}$ analytical model for the angular rate sensor triad error are identical to steps leading to the $\delta \underline{a}_{SF}^{B}$ accelerometer triad error analytical model, beginning with Equations (8.1.1.2-2), (8.1.1.2-3) and (8.1.1.2-5) - (8.1.1.2-7). For this development we will assume that the accelerometer anisoinertia and size effect compensations utilized are accurate enough that the residual error after compensation in Equation (8.1.1.2-3) is negligible. With this approximation, the equivalent to Equation (12.4-12) for the accelerometer triad becomes:

$$\delta \underline{a}_{SF}^{B} = \delta L_{Scal/Mis}\, \underline{a}_{SF}^{B} + \delta \underline{L}_{Bias} + \delta \underline{a}_{Quant} + \delta \underline{a}_{Rand} \tag{12.4-14}$$

with

$$\delta L_{Scal} \equiv A_{WtC}\, A_{Wt}^{-1} - I \qquad \delta L_{Mis} \equiv L_{Mis} - L_{MisC}$$

$$\delta L_{Scal/Mis} \equiv \delta L_{Scal} + \delta L_{Mis} \tag{12.4-15}$$

$$\delta \underline{L}_{Bias} \equiv \underline{L}_{Bias} - \underline{L}_{BiasC} \qquad \delta \underline{a}_{Quant} \equiv \partial \underline{a}_{Quant} - \partial \underline{a}_{QuantC}$$

where

A_{Wt} = Idealized accelerometer triad scale factor diagonal matrix.

$\underline{L}_{Bias}$ = Idealized accelerometer triad effective bias compensation vector.

L_{Mis} = Idealized accelerometer effective misalignment compensation matrix.

$\partial \underline{a}_{Quant}$ = Uncompensated accelerometer triad pulse quantization error vector. Note that the quantization error is denoted here as $\partial \underline{a}_{Quant}$ while in Section 8.1.1.2 it is denoted as $\delta \underline{a}_{Quant}$. The distinction has been made here to differentiate errors that are partially correctable by compensation (the $\partial \underline{a}_{Quant}$ term) from residual errors remaining after compensation has been applied (the $\delta \underline{a}_{Quant}$ term).

$\delta \underline{a}_{Rand}$ = Accelerometer triad random error vector.

$A_{Wt\,C}$, L_{MisC}, $\underline{L}_{BiasC}$, $\partial\underline{a}_{QuantC}$ = Values for A_{Wt}, L_{Mis}, $\underline{L}_{Bias}$, $\partial\underline{a}_{Quant}$ utilized for accelerometer compensation in the strapdown system computer.

δL_{Mis}, $\delta\underline{L}_{Bias}$, $\delta\underline{a}_{Quant}$ = Residual errors in L_{Mis}, $\underline{L}_{Bias}$, $\partial\underline{a}_{Quant}$ remaining after applying compensation terms L_{MisC}, $\underline{L}_{BiasC}$, $\partial\underline{a}_{QuantC}$.

δL_{Scal} = Normalized residual scale factor error remaining in A_{Wt} after applying the $A_{Wt\,C}$ scale factor matrix.

$\delta L_{Scal/Mis}$ = Effective residual scale-factor/misalignment error matrix remaining in A_{Wt} and L_{Mis} after applying compensation terms $A_{Wt\,C}$ and L_{MisC}.

Equation (12.4-14) is the analytical model for $\delta\underline{a}_{SF}^{B}$ in N Frame navigation error Equations (12.3.3-6), (12.3.4-39), (12.3.5-29) and (12.3.7.2-6). The analytical models used for $\delta L_{Scal/Mis}$ and $\delta\underline{L}_{Bias}$ in (12.4-14) depend on the accelerometer selected, its particular error characteristics and the associated compensation terms. The $\delta\underline{a}_{Rand}$ term is typically modeled as white noise. The $\delta\underline{a}_{Quant}$ term is typically represented as the time derivative of white noise which requires some revision in the navigation error equations (discussed in Section 12.5) for compatibility with standard random process equation formats (e.g., error state dynamic equation format discussed in Section 15.1 in which noise terms are represented as white noise).

12.5 ERROR EQUATION REVISIONS TO ENHANCE QUANTIZATION NOISE MODELING

The $\partial\underline{\omega}_{Quant}$ and $\partial\underline{a}_{Quant}$ terms in strapdown inertial sensor error model Equations (12.4-10) and (12.4-15) represent the quantization of the output pulse format from the angular rate sensors and accelerometers. Each output pulse from a sensor signifies to the strapdown computer that the integral of the sensor input (integrated angular rate or specific force acceleration) has changed by a specified pulse size from the last output pulse. At a constant input to the sensor the integrated sensor input is a linear ramp in time, hence, the output is a constant frequency pulse train. Under dynamic input conditions, the pulse rate varies from instant to instant.

In Chapter 7 (and Chapter 19, Section 19.1) we found that the sensor inputs utilized in the strapdown computer are integrated samples of the sensor outputs over each computer update cycle. In practice, this is achieved by counting the sensor output pulses. We can imagine the attitude and velocity parameters in the strapdown computer being generated as a repetitive summing operation on the sensor count samples. For perfect inertial sensors, the complete integrals so generated will be correct, but only to within a pulse, because the instantaneous pulse output will in general, not occur exactly at the computer sample time instant. The associated error effect is called pulse quantization which can be accurately modeled as a white uniform random error process on each integrated sensor output. The $\partial\underline{\omega}_{Quant}$ and $\partial\underline{a}_{Quant}$ terms in Equations (12.4-10) and (12.4-15) represent instantaneous random errors on the direct (not integrated) sensor outputs. Hence, $\partial\underline{\omega}_{Quant}$ and $\partial\underline{a}_{Quant}$ represent the time derivative of the white random quantization error process associated with the integrated output. The $\delta\underline{\omega}_{Quant}$ and $\delta\underline{a}_{Quant}$ terms in (12.4-12) and (12.4-14) represent residual errors in $\partial\underline{\omega}_{Quant}$ and $\partial\underline{a}_{Quant}$ after applying quantization compensation (if any). Thus, $\delta\underline{\omega}_{Quant}$ and $\delta\underline{a}_{Quant}$ also represent the time derivative of white noise processes. Because a white uniform random process on the compensated integrated sensor outputs is easily defined mathematically, it is expeditious to substitute for $\delta\underline{\omega}_{Quant}$ and $\delta\underline{a}_{Quant}$:

$$\delta\underline{\omega}_{Quant} = \delta\dot{\underline{\alpha}}_{Quant} \qquad \delta\underline{a}_{Quant} = \delta\dot{\underline{\upsilon}}_{Quant} \tag{12.5-1}$$

where

$\delta\underline{\alpha}_{Quant}, \delta\underline{\upsilon}_{Quant}$ = Angular rate sensor and accelerometer white noise quantization error effects associated with the quantization compensated integral of angular rate sensor and accelerometer outputs.

Using $\delta\underline{\omega}_{Quant}$ from Equations (12.5-1), let us substitute the Equation (12.4-12) $\delta\underline{\omega}_{IB}^{B}$ strapdown angular rate sensor triad error model into the Equations (12.3.4-39) $\dot{\underline{\gamma}}^{N}$ expression to obtain after rearrangement:

$$\begin{aligned}\dot{\underline{\gamma}}^{N} + C_B^N\, \delta\dot{\underline{\alpha}}_{Quant} = &- C_B^N \left(\delta K_{Scal/Mis}\, \underline{\omega}_{IB}^{B} + \delta\underline{K}_{Bias} + \delta\underline{\omega}_{Rand}\right) \\ &- \underline{\omega}_{IN}^{N} \times \underline{\gamma}^{N} + \underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N} + \delta\underline{\omega}_{EN}^{N}\end{aligned} \tag{12.5-2}$$

We also note that:

$$C_B^N\, \delta\dot{\underline{\alpha}}_{Quant} = \frac{d}{dt}\left(C_B^N\, \delta\underline{\alpha}_{Quant}\right) - \dot{C}_B^N\, \delta\underline{\alpha}_{Quant} \tag{12.5-3}$$

so that (12.5-2) is equivalently:

$$\begin{aligned}\frac{d}{dt}\left(\underline{\gamma}^{N} + C_B^N\, \delta\underline{\alpha}_{Quant}\right) = &- C_B^N \left(\delta K_{Scal/Mis}\, \underline{\omega}_{IB}^{B} + \delta\underline{K}_{Bias} + \delta\underline{\omega}_{Rand}\right) + \dot{C}_B^N\, \delta\underline{\alpha}_{Quant} \\ &- \underline{\omega}_{IN}^{N} \times \underline{\gamma}^{N} + \underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N} + \delta\underline{\omega}_{EN}^{N}\end{aligned} \tag{12.5-4}$$

The form of (12.5-4) suggests the definition of revised attitude and angular rate sensor error parameters:

$$\underline{\gamma}^{N*} \equiv \underline{\gamma}^N + C_B^N\, \delta\underline{\alpha}_{Quant} \tag{12.5-5}$$

$$\delta\underline{\omega}_{IB}^{B*} \equiv \delta K_{Scal/Mis}\, \underline{\omega}_{IB}^B + \delta\underline{K}_{Bias} + \delta\underline{\omega}_{Rand} \tag{12.5-6}$$

where

$\delta\underline{\omega}_{IB}^{B*}$ = Angular rate sensor error exclusive of angular rate sensor quantization error effects.

$\underline{\gamma}^{N*}$ = Revised form of $\underline{\gamma}^N$ that neglects the $C_B^N\, \delta\underline{\alpha}_{Quant}$ part of the $\underline{\gamma}^N$ quantization error (which is the dominant quantization term affecting $\underline{\gamma}^N$). As will soon be apparent from Equation (12.5-8) to follow, $\underline{\gamma}^{N*}$ still contains angular rate sensor quantization error input under applied angular rates.

The converse of Equation (12.5-5) will also be useful:

$$\underline{\gamma}^N = \underline{\gamma}^{N*} - C_B^N\, \delta\underline{\alpha}_{Quant} \tag{12.5-7}$$

Substitution of (12.5-5) - (12.5-7) into (12.5-4) gives:

$$\begin{aligned}\dot{\underline{\gamma}}^{N*} = {} & - C_B^N\, \delta\underline{\omega}_{IB}^{B*} - \underline{\omega}_{IN}^N \times \underline{\gamma}^{N*} + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N \\ & + \left[\dot{C}_B^N + \left(\underline{\omega}_{IN}^N \times\right) C_B^N\right] \delta\underline{\alpha}_{Quant}\end{aligned} \tag{12.5-8}$$

The $\left[\dot{C}_B^N + \left(\underline{\omega}_{IN}^N \times\right) C_B^N\right]$ term in (12.5-8) can be reduced using $\dot{C}_B^N$ from Equations (12.1.2-6):

$$\dot{C}_B^N + \left(\underline{\omega}_{IN}^N \times\right) C_B^N = \left[C_B^N \left(\underline{\omega}_{IB}^B \times\right) - \left(\underline{\omega}_{IN}^N \times\right) C_B^N\right] + \left(\underline{\omega}_{IN}^N \times\right) C_B^N = C_B^N \left(\underline{\omega}_{IB}^B \times\right) \tag{12.5-9}$$

Substituting (12.5-9) into (12.5-8) then yields:

$$\dot{\underline{\gamma}}^{N*} = - C_B^N\, \delta\underline{\omega}_{IB}^{B*} - \underline{\omega}_{IN}^N \times \underline{\gamma}^{N*} + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N + C_B^N \left(\underline{\omega}_{IB}^B \times \delta\underline{\alpha}_{Quant}\right) \tag{12.5-10}$$

Note in Equation (12.5-10) that only white angular rate sensor noise appears (i.e., $\delta\underline{\alpha}_{Quant}$ and $\delta\underline{\omega}_{Rand}$ in $\delta\underline{\omega}_{IB}^{B*}$) and no angular rate sensor white noise derivatives.

Encouraged by these results we apply the identical procedure to the velocity and accelerometer error terms in Equations (12.3.4-39) and (12.4-14), viz.:

$$\delta\underline{v}^{N*} \equiv \delta\underline{v}^{N} - C_B^N\, \delta\underline{\upsilon}_{Quant} \tag{12.5-11}$$

$$\delta\underline{a}_{SF}^{B*} \equiv \delta L_{Scal/Mis}\, \underline{a}_{SF}^{B} + \delta\underline{L}_{Bias} + \delta\underline{a}_{Rand} \tag{12.5-12}$$

$$\delta\underline{v}^{N} = \delta\underline{v}^{N*} + C_B^N\, \delta\underline{\upsilon}_{Quant} \tag{12.5-13}$$

where

$\delta\underline{a}_{SF}^{B*}$ = Accelerometer error exclusive of accelerometer quantization error effects.

$\delta\underline{v}^{N*}$ = Revised form of $\delta\underline{v}^{N}$ that neglects the $C_B^N\, \delta\underline{\upsilon}_{Quant}$ part of the $\delta\underline{v}^{N}$ quantization error (which is the dominant quantization term affecting $\delta\underline{v}^{N}$). As will soon be apparent from Equation (12.5-15) to follow, $\delta\underline{v}^{N*}$ still contains accelerometer quantization error input under applied angular rates.

The derivative of (12.5-11) is:

$$\delta\dot{\underline{v}}^{N*} = \delta\dot{\underline{v}}^{N} - \dot{C}_B^N\, \delta\underline{\upsilon}_{Quant} - C_B^N\, \delta\dot{\underline{\upsilon}}_{Quant} \tag{12.5-14}$$

Substitution for $\delta\dot{\underline{v}}^{N}$ with $\delta\underline{\omega}_{EN}^{N}$ from Equations (12.3.4-39) and the Equation (12.4-14) $\delta\underline{a}_{SF}^{B}$ accelerometer error model into (12.5-14) with (12.5-12), substitution for $\delta\underline{a}_{Quant}$ from Equations (12.5-1), substitution for $\underline{\gamma}^{N}$, $\delta\underline{v}^{N}$ from Equations (12.5-7) and (12.5-13), and applying generalized Equation (3.1.1-15) then yields for $\delta\dot{\underline{v}}^{N*}$:

$$\begin{aligned}
\delta\dot{\underline{v}}^{N*} = \; & C_B^N\, \delta\underline{a}_{SF}^{B*} + \underline{a}_{SF}^{N} \times \underline{\gamma}^{N*} + \underline{v}^{N} \times \left(\delta\underline{\omega}_{EN}^{N*} + 2\, \underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N}\right) \\
& - \left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times \delta\underline{v}^{N*} + \delta\underline{g}_{Mdl}^{N} \\
& + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta h + C_2\, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^{N} \\
& - \left(\underline{a}_{SF}^{N} \times\right) C_B^N\, \delta\underline{\alpha}_{Quant} + \left\{\frac{1}{r_l} \left(\underline{v}^{N} \times\right) \left(\underline{u}_{ZN}^{N} \times\right) C_B^N \right. \\
& \left. - \left[\left(\underline{\omega}_{EN}^{N} + 2\, \underline{\omega}_{IE}^{N}\right) \times\right] C_B^N - \dot{C}_B^N \right\} \delta\underline{\upsilon}_{Quant}
\end{aligned} \tag{12.5-15}$$

$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$

with

$$\delta\underline{\omega}_{EN}^{N*} \equiv \delta\rho_{ZN}\,\underline{u}_{ZN}^{N} + \frac{1}{r_l}\left(\underline{u}_{ZN}^{N} \times \delta\underline{v}^{N*}\right) - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^{N} \times \underline{v}^{N}\right)\delta h \tag{12.5-16}$$

The $-\left[\left(\underline{\omega}_{EN}^{N} + 2\,\underline{\omega}_{IE}^{N}\right)\times\right] C_B^N - \dot{C}_B^N$ term in (12.5-15) can be expanded using $\dot{C}_B^N$ from Equations (12.1.2-6) with (12.3.6.1-4) for $\underline{\omega}_{IN}^{N}$ and generalized Equation (3.1.1-38):

$$\begin{aligned}
-\left[\left(\underline{\omega}_{EN}^{N} + 2\,\underline{\omega}_{IE}^{N}\right)\times\right] C_B^N - \dot{C}_B^N &= -\left(\underline{\omega}_{EN}^{N}\times\right) C_B^N - 2\left(\underline{\omega}_{IE}^{N}\times\right) C_B^N \\
&\quad - \left\{C_B^N\left(\underline{\omega}_{IB}^{B}\times\right) - \left[\left(\underline{\omega}_{IE}^{N}\times\right) + \left(\underline{\omega}_{EN}^{N}\times\right)\right] C_B^N\right\} \\
&= -\,C_B^N\left(\underline{\omega}_{IB}^{B}\times\right) - \left(\underline{\omega}_{IE}^{N}\times\right) C_B^N = -\left[C_B^N\left(\underline{\omega}_{IB}^{B}\times\right)\left(C_B^N\right)^T + \left(\underline{\omega}_{IE}^{N}\times\right)\right] C_B^N \\
&= -\left\{\left[\left(C_B^N\,\underline{\omega}_{IB}^{B}\right)\times\right] + \left(\underline{\omega}_{IE}^{N}\times\right)\right\} C_B^N = -\left[\left(C_B^N\,\underline{\omega}_{IB}^{B} + \underline{\omega}_{IE}^{N}\right)\times\right] C_B^N
\end{aligned} \tag{12.5-17}$$

Using (12.5-17) in (12.5-15) then obtains the final form for $\delta\dot{\underline{v}}^{N*}$:

$$\begin{aligned}
\delta\dot{\underline{v}}^{N*} &= C_B^N\,\delta\underline{a}_{SF}^{B*} + \underline{a}_{SF}^{N} \times \underline{\gamma}^{N*} + \underline{v}^{N} \times \left(\delta\underline{\omega}_{EN}^{N*} + 2\,\underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N}\right) \\
&\quad - \left(\underline{\omega}_{EN}^{N} + 2\,\underline{\omega}_{IE}^{N}\right) \times \delta\underline{v}^{N*} + \delta\underline{g}_{Mdl}^{N} \\
&\quad + \left[\left(F(h)\,\frac{g}{R} - C_2\right)\delta h + C_2\,\delta h_{Prsr} - \delta e_{vc_3}\right]\underline{u}_{ZN}^{N} \\
&\quad - \left(\underline{a}_{SF}^{N}\times\right) C_B^N\,\delta\underline{\alpha}_{Quant} + \left\{\frac{1}{r_l}\left(\underline{v}^{N}\times\right)\left(\underline{u}_{ZN}^{N}\times\right) - \left[\left(C_B^N\underline{\omega}_{IB}^{B} + \underline{\omega}_{IE}^{N}\right)\times\right]\right\} C_B^N\,\delta\underline{\upsilon}_{Quant}
\end{aligned} \tag{12.5-18}$$

$$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$$

Note in Equation (12.5-18) (with (12.5-12) and (12.5-16) for $\delta\underline{a}_{SF}^{B*}$, $\delta\underline{\omega}_{EN}^{N*}$) that all noise terms are white in accordance with typical standard analytical formats.

The Equation (12.5-16) $\delta\underline{\omega}_{EN}^{N*}$ form can also be utilized in $\dot{\underline{\gamma}}^{N*}$ Equation (12.5-10) by substituting for $\delta\underline{\omega}_{EN}^{N}$ from Equations (12.3.4-39) and using $\delta\underline{v}^{N}$ from (12.5-13). The result is:

$$\begin{aligned}
\dot{\underline{\gamma}}^{N*} &= -\,C_B^N\,\delta\underline{\omega}_{IB}^{B*} - \underline{\omega}_{IN}^{N} \times \underline{\gamma}^{N*} + \underline{\omega}_{IE}^{N} \times \underline{\varepsilon}^{N} + \delta\underline{\omega}_{EN}^{N*} \\
&\quad + C_B^N\left(\underline{\omega}_{IB}^{B} \times \delta\underline{\alpha}_{Quant}\right) + \frac{1}{r_l}\,\underline{u}_{ZN}^{N} \times \left(C_B^N\,\delta\underline{\upsilon}_{Quant}\right)
\end{aligned} \tag{12.5-19}$$

Note in Equation (12.5-19) (with (12.5-6) and (12.5-16) for $\delta\underline{\omega}_{IB}^{B}*$, $\delta\underline{\omega}_{EN}^{N}*$) that all noise terms are now white in accordance with typical standard analytical formats.

The $\dot{\underline{\varepsilon}}^{N}$ and $\delta\dot{h}$ terms in (12.3.4-39) are revised similarly:

$$
\begin{aligned}
\dot{\underline{\varepsilon}}^{N} &= -\,\underline{\omega}_{EN}^{N} \times \underline{\varepsilon}^{N} + \delta\underline{\omega}_{EN}^{N}* + \frac{1}{r_l}\,\underline{u}_{ZN}^{N} \times \left(C_B^N\,\delta\underline{\upsilon}_{Quant}\right) \\
\delta\dot{h} &= \underline{u}_{ZN}^{N} \cdot \delta\underline{v}^{N}* - C_3\left(\delta h - \delta h_{Prsr}\right) + \underline{u}_{ZN}^{N} \cdot \left(C_B^N\,\delta\underline{\upsilon}_{Quant}\right)
\end{aligned}
\qquad (12.5\text{-}20)
$$

Equations (12.5-18) - (12.5-20) with (12.5-6), (12.5-12) and (12.5-16) are a complete consistent set for the new (and old) variables $\underline{\gamma}^{N}*$, $\delta\underline{v}^{N}*$, $\underline{\varepsilon}^{N}$, δh that only contain white type noise terms. These equations can be used in place of navigation error Equations (12.3.4-39) for the $\underline{\gamma}^{N}$, $\delta\underline{v}^{N}$, $\underline{\varepsilon}^{N}$, δh error parameters.

The identical procedure can be utilized to obtain revised forms of navigation error Equations (12.3.3-6), (12.3.5-29) and (12.3.7.2-6) for the $\left(\underline{\psi}^{N}, \delta\underline{V}^{N}, \delta\underline{R}^{N}\right)$, $\left(\underline{\gamma}^{N}, \delta\underline{V}^{N}, \delta\underline{R}^{N}, \varepsilon_{ZN}\right)$ and $\left(\underline{\psi}^{N}, \delta\underline{\upsilon}^{N}, \delta\underline{R}^{N}\right)$ error parameter sets. As above, we first define revised forms as follows:

$$\underline{\psi}^{N}* \equiv \underline{\psi}^{N} + C_B^N\,\delta\underline{\alpha}_{Quant} \qquad (12.5\text{-}21)$$

$$\delta\underline{V}^{N}* \equiv \delta\underline{V}^{N} - C_B^N\,\delta\underline{\upsilon}_{Quant} \qquad (12.5\text{-}22)$$

$$\delta\underline{\upsilon}^{N}* \equiv \delta\underline{\upsilon}^{N} - C_B^N\,\delta\underline{\upsilon}_{Quant} \qquad (12.5\text{-}23)$$

and the converse:

$$\underline{\psi}^{N} = \underline{\psi}^{N}* - C_B^N\,\delta\underline{\alpha}_{Quant} \qquad (12.5\text{-}24)$$

$$\delta\underline{V}^{N} = \delta\underline{V}^{N}* + C_B^N\,\delta\underline{\upsilon}_{Quant} \qquad (12.5\text{-}25)$$

$$\delta\underline{\upsilon}^{N} = \delta\underline{\upsilon}^{N}* + C_B^N\,\delta\underline{\upsilon}_{Quant} \qquad (12.5\text{-}26)$$

where

$\underline{\psi}^{N}*$ = Revised form of $\underline{\psi}^{N}$ that neglects the $C_B^N\,\delta\underline{\alpha}_{Quant}$ part of the $\underline{\psi}^{N}$ quantization error (which is the dominant quantization term affecting $\underline{\psi}^{N}$).

$\delta \underline{V}^{N*}$, $\delta \underline{\upsilon}^{N*}$ = Revised forms of $\delta \underline{V}^N$, $\delta \underline{\upsilon}^N$ that neglect the $C_B^N \, \delta \underline{\upsilon}_{Quant}$ part of the $\delta \underline{V}^N$, $\delta \underline{\upsilon}^N$ quantization errors (which is the dominant quantization term affecting $\delta \underline{V}^N$, $\delta \underline{\upsilon}^N$).

The equivalency relationships between the new error parameters (with the asterisk *) is easily obtained by substituting (12.5-7), (12.5-13) and (12.5-24) - (12.5-26) into equivalency Equations (12.2.1-17), (12.2.2-5) and (12.2.2-16). Not surprisingly, the results are:

$$\underline{\psi}^{N*} = \underline{\gamma}^{N*} - \underline{\varepsilon}^N \tag{12.5-27}$$

$$\delta \underline{V}^{N*} = \delta \underline{v}^{N*} + \underline{\varepsilon}^N \times \underline{v}^N \tag{12.5-28}$$

$$\delta \underline{\upsilon}^{N*} = \delta \underline{V}^{N*} + \underline{\omega}_{IE}^N \times \delta \underline{R}^N \tag{12.5-29}$$

We see then from Equations (12.5-27) - (12.5-29) that the form of the equivalency equations for the new $\underline{\gamma}^{N*}$, $\underline{\psi}^{N*}$, $\delta \underline{v}^{N*}$, $\delta \underline{V}^{N*}$, $\delta \underline{\upsilon}^{N*}$ parameters is identical to Equations (12.2.1-17), (12.2.2-5) and (12.2.2-16) for the original $\underline{\gamma}^N$, $\underline{\psi}^N$, $\delta \underline{v}^N$, $\delta \underline{V}^N$, $\delta \underline{\upsilon}^N$ parameters. This leads us to make the important generalization that we can convert the original error equations to the new asterisk * versions, and then delete the asterisk * notation for simplicity. The $\underline{\gamma}^N$, $\underline{\psi}^N$, $\delta \underline{v}^N$, $\delta \underline{V}^N$, $\delta \underline{\upsilon}^N$ parameters so created will then be understood to not include quantization noise and the original equivalency Equations (12.2.1-17), (12.2.2-5) and (12.2.2-16) will remain applicable. We must understand, however, that any other error parameters derived from this set (such as the attitude Euler angle errors $\delta\phi$, $\delta\theta$, $\delta\psi_P$ in Equations (12.2.1-38)) will also not include quantization noise. Based on this generalization and the development procedure utilized above, the following subsections summarize versions of navigation error Equations (12.3.3-6), (12.3.4-39), (12.3.5-29) and (12.3.7.2-6) for the $\left(\underline{\psi}^N, \delta \underline{V}^N, \delta \underline{R}^N\right)$, $\left(\underline{\gamma}^N, \delta \underline{v}^N, \underline{\varepsilon}^N, \delta h\right)$, $\left(\underline{\gamma}^N, \delta \underline{V}^N, \delta \underline{R}^N, \varepsilon_{ZN}\right)$ and $\left(\underline{\psi}^N, \delta \underline{\upsilon}^N, \delta \underline{R}^N\right)$ error parameter sets that have been revised to exclude strapdown inertial sensor quantization noise in the $\underline{\gamma}^N$, $\underline{\psi}^N$, $\delta \underline{v}^N$, $\delta \underline{V}^N$, $\delta \underline{\upsilon}^N$ error parameters. The following subsections will also discuss the structure of the typical error rate equations for the angular rate sensor and accelerometer error parameters.

12.5.1 REVISED ERROR RATE EQUATIONS FOR E FRAME DEFINED ERROR PARAMETERS PROJECTED ONTO THE N FRAME

The navigation error equations defined in the E Frame utilize the $\underline{\psi}$, $\delta\underline{V}$, $\delta\underline{R}$ error parameters. Equations (12.3.3-6) summarize the rate equations for the $\underline{\psi}$, $\delta\underline{V}$, $\delta\underline{R}$ parameters projected in the N Frame. Equations for the strapdown inertial sensor errors in (12.3.3-6) are given by Equations (12.4-12), (12.4-14) and (12.5-1). Following the development procedure used in Section 12.5 for the $\underline{\gamma}$, $\delta\underline{v}$, $\underline{\varepsilon}$, δh parameters, a revised set of error equations can be developed from Equations (12.3.3-6) in terms of the revised parameters $\underline{\psi}^{*N}$, $\delta\underline{V}^{*N}$, $\delta\underline{R}^{N}$ of which $\underline{\psi}^{*N}$, $\delta\underline{V}^{*N}$ are defined by Equations (12.5-21) and (12.5-22) as not containing sensor quantization noise. Following the generalization plan in Section 12.5, the revised $\underline{\psi}^{*N}$, $\delta\underline{V}^{*N}$, $\delta\underline{R}^{N}$ rate equations can then be simplified by deleting the asterisk * notation. Such a revised $\underline{\psi}$, $\delta\underline{V}$, $\delta\underline{R}$ equation set projected in the N Frame (revised from Equations (12.3.3-6)) is summarized below by Equations (12.5.1-1) together with remaining applicable elements of (12.3.3-6). The $\underline{\psi}^{N}$, $\delta\underline{V}^{N}$ parameters in Equations (12.5.1-1) then, are the $\underline{\psi}^{*N}$, $\delta\underline{V}^{*N}$ terms defined by Equations (12.5-21) and (12.5-22) with quantization noise removed, but with the asterisk * notation deleted for simplicity (not to be confused with the $\underline{\psi}^{N}$, $\delta\underline{V}^{N}$ parameters in Equations (12.3.3-6) which include quantization noise error). The equivalency relationships in Sections 12.2.1 and 12.2.2 between the $\underline{\gamma}$, $\underline{\psi}$, $\delta\underline{v}$, $\delta\underline{V}$, $\delta\underline{\upsilon}$ parameters in Equations (12.5.1-1) and subsequent Equations (12.5.2-1), (12.5.3-1) and (12.5.4-1) remain valid as discussed in Section 12.5.

$$
\begin{aligned}
\dot{\underline{\psi}}^{N} &= - C_B^N \, \delta\underline{\omega}_{IB}^{B} - \underline{\omega}_{IN}^{N} \times \underline{\psi}^{N} + C_B^N \left(\underline{\omega}_{IB}^{B} \times \delta\underline{\alpha}_{Quant}\right) \\
\delta\underline{\omega}_{IB}^{B} &= \delta K_{Scal/Mis} \, \underline{\omega}_{IB}^{B} + \delta\underline{K}_{Bias} + \delta\underline{\omega}_{Rand} \\
\delta\dot{\underline{V}}^{N} &= C_B^N \, \delta\underline{a}_{SF}^{B} + \underline{a}_{SF}^{N} \times \underline{\psi}^{N} - \frac{g}{R} \, \delta\underline{R}_{H}^{N} - \left(\underline{\omega}_{IE}^{N} + \underline{\omega}_{IN}^{N}\right) \times \delta\underline{V}^{N} + \delta\underline{g}_{Mdl}^{N} \\
&\quad + \left[\left(F(h) \, \frac{g}{R} - C_2\right) \delta R + C_2 \, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^{N} \qquad (12.5.1\text{-}1) \\
&\quad - \left(\underline{a}_{SF}^{N} \times\right) C_B^N \, \delta\underline{\alpha}_{Quant} - \left[\left(C_B^N \, \underline{\omega}_{IB}^{B} + \underline{\omega}_{IE}^{N}\right) \times\right] C_B^N \, \delta\underline{\upsilon}_{Quant}
\end{aligned}
$$

$F(h) = 2 \quad \text{For } h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For } h < 0$

$$\delta\underline{a}_{SF}^{B} = \delta L_{Scal/Mis} \, \underline{a}_{SF}^{B} + \delta\underline{L}_{Bias} + \delta\underline{a}_{Rand}$$

(Continued)

$$\delta \dot{\underline{R}}^N = \delta \underline{V}^N - \underline{\omega}_{EN}^N \times \delta \underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr} \right) \underline{u}_{ZN}^N + C_B^N \, \delta \underline{\upsilon}_{Quant}$$

$$\delta \dot{e}_{vc_3} = C_1 \left(\delta R - \delta h_{Prsr} \right)$$

$$\delta \underline{R}_H^N = \delta \underline{R}^N - \delta R \, \underline{u}_{ZN}^N \qquad \text{(12.5.1-1) (Continued)}$$

$$\delta R = \underline{u}_{ZN}^N \cdot \delta \underline{R}^N$$

12.5.2 REVISED ERROR RATE EQUATIONS FOR N FRAME DEFINED ERROR PARAMETERS PROJECTED ONTO THE N FRAME

The navigation error equations defined in the N Frame utilize the $\underline{\gamma}$, $\delta\underline{v}$, $\underline{\varepsilon}$, δh error parameters. Following the generalization plan in Section 12.5 (i.e., deleting the asterisk * notation), the revised $\underline{\gamma}$, $\delta\underline{v}$, $\underline{\varepsilon}$, δh rate equations in the N Frame (revised from Equations (12.3.4-39)) which have been derived in Section 12.5 (as Equations (12.5-18) - (12.5-20), (12.5-6), (12.5-12), (12.5-16)), are summarized below by Equations (12.5.2-1), together with remaining applicable elements of (12.3.4-39) . The $\underline{\gamma}^N$, $\delta\underline{v}^N$ parameters in Equations (12.5.2-1) are the $\underline{\gamma}^{*N}$, $\delta\underline{v}^{*N}$ terms defined by Equations (12.5-5) and (12.5-11) with quantization noise removed, but with the asterisk * notation deleted for simplicity (not to be confused with the $\underline{\gamma}^N$, $\delta\underline{v}^N$ parameters in Equations (12.3.4-39) which include quantization noise error). The equivalency relationships in Sections 12.2.1 and 12.2.2 between the $\underline{\gamma}$, $\underline{\psi}$, $\delta\underline{v}$, $\delta\underline{V}$, $\delta\underline{\upsilon}$ parameters in Equations (12.5.1-1), (12.5.2-1) and subsequent Equations (12.5.3-1), (12.5.4-1) remain valid as discussed in Section 12.5.

$$\begin{aligned} \dot{\underline{\gamma}}^N = & - C_B^N \, \delta \underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \underline{\omega}_{IE}^N \times \underline{\varepsilon}^N + \delta \underline{\omega}_{EN}^N \\ & + C_B^N \left(\underline{\omega}_{IB}^B \times \delta \underline{\alpha}_{Quant} \right) + \frac{1}{r_l} \, \underline{u}_{ZN}^N \times \left(C_B^N \, \delta \underline{\upsilon}_{Quant} \right) \end{aligned} \qquad (12.5.2\text{-}1)$$

$$\delta \underline{\omega}_{IB}^B = \delta K_{Scal/Mis} \, \underline{\omega}_{IB}^B + \delta \underline{K}_{Bias} + \delta \underline{\omega}_{Rand}$$

(Continued)

$$\begin{aligned}\delta\dot{\underline{v}}^N = \; & C_B^N \, \delta \underline{a}_{SF}{}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N + \underline{v}^N \times \left(\delta\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \\ & - \left(\underline{\omega}_{EN}^N + 2\,\underline{\omega}_{IE}^N\right) \times \delta\underline{v}^N + \delta\underline{g}_{Mdl}^N \\ & + \left[\left(F(h)\,\frac{g}{R} - C_2\right)\delta h + C_2\,\delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^N \\ & - \left(\underline{a}_{SF}^N \times\right) C_B^N \, \delta\underline{\alpha}_{Quant} \\ & + \left\{\frac{1}{r_l}\left(\underline{v}^N \times\right)\left(\underline{u}_{ZN}^N \times\right) - \left[\left(C_B^N \underline{\omega}_{IB}^B + \underline{\omega}_{IE}^N\right)\times\right]\right\} C_B^N \, \delta\underline{\upsilon}_{Quant}\end{aligned} \qquad \begin{matrix}(12.5.2\text{-}1)\\ (\text{Continued})\end{matrix}$$

$$F(h) = 2 \quad \text{For} \;\; h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \;\; h < 0$$

$$\delta\underline{a}_{SF}^B = \delta L_{Scal/Mis}\, \underline{a}_{SF}^B + \delta\underline{L}_{Bias} + \delta\underline{a}_{Rand}$$

$$\dot{\underline{\varepsilon}}^N = - \underline{\omega}_{EN}^N \times \underline{\varepsilon}^N + \delta\underline{\omega}_{EN}^N + \frac{1}{r_l}\,\underline{u}_{ZN}^N \times \left(C_B^N \, \delta\underline{\upsilon}_{Quant}\right)$$

$$\delta\dot{h} = \underline{u}_{ZN}^N \cdot \delta\underline{v}^N - C_3\left(\delta h - \delta h_{Prsr}\right) + \underline{u}_{ZN}^N \cdot \left(C_B^N \, \delta\underline{\upsilon}_{Quant}\right)$$

$$\delta\dot{e}_{vc_3} = C_1\left(\delta h - \delta h_{Prsr}\right)$$

$$\delta\underline{\omega}_{EN}^N = \delta\rho_{ZN}\,\underline{u}_{ZN}^N + \frac{1}{r_l}\left(\underline{u}_{ZN}^N \times \delta\underline{v}^N\right) - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^N \times \underline{v}^N\right)\delta h$$

$$\delta\rho_{ZN} = 0 \qquad \text{For Wander Azimuth Implementation}$$

$$\delta\rho_{ZN} = -\left(\underline{\omega}_{IE}^N \times \underline{\varepsilon}^N\right) \cdot \underline{u}_{ZN}^N \qquad \text{For Free Azimuth Implementation}$$

12.5.3 <u>REVISED ERROR RATE EQUATIONS FOR MIXED E AND N FRAME DEFINED ERROR PARAMETERS PROJECTED ONTO THE N FRAME</u>

The mixed E and N Frame defined navigation error equations utilize the $\underline{\gamma}$, $\delta\underline{V}$, $\delta\underline{R}$, ε_{ZN} error parameters. Equations (12.3.5-29) summarize the rate equations for the $\underline{\gamma}$, $\delta\underline{V}$, $\delta\underline{R}$, ε_{ZN} parameters projected in the N Frame, . The equations for the strapdown inertial sensor errors in Equations (12.3.5-29) are given by Equations (12.4-12), (12.4-14) and (12.5-1). Following the development procedure used in Section 12.5 for the $\underline{\gamma}$, $\delta\underline{v}$, $\underline{\varepsilon}$, δh parameters, a revised set of error equations can be developed from Equations (12.3.5-29) in terms of the revised parameters $\underline{\gamma}^{*N}$, $\delta\underline{V}^{*N}$, $\delta\underline{R}^N$, ε_{ZN} of which $\underline{\gamma}^{*N}$, $\delta\underline{V}^{*N}$ are defined by Equations (12.5-5) and (12.5-22) as not containing sensor quantization noise. Following the generalization plan in Section 12.5, the revised $\underline{\gamma}^{*N}$, $\delta\underline{V}^{*N}$, $\delta\underline{R}^N$, ε_{ZN} rate equations can then be simplified by deleting the asterisk * notation. Such a revised $\underline{\gamma}$, $\delta\underline{V}$, $\delta\underline{R}$, ε_{ZN} equation set projected in the N Frame (revised from Equations (12.3.5-29)) is summarized below by Equations (12.5.3-1) together with remaining applicable elements of (12.3.5-29). The $\underline{\gamma}$, $\delta\underline{V}$ parameters in Equations (12.5.3-1) then, are the $\underline{\gamma}^{*N}$, $\delta\underline{V}^{*N}$ terms defined by Equations (12.5-5) and (12.5-22) with quantization noise removed, but with the asterisk * notation deleted for simplicity (not to be confused with the $\underline{\gamma}$, $\delta\underline{V}$ parameters in Equations (12.3.5-29) which include quantization noise error). The equivalency relationships in Sections 12.2.1 and 12.2.2 between the $\underline{\gamma}$, $\underline{\psi}$, $\delta\underline{v}$, $\delta\underline{V}$, $\delta\underline{\upsilon}$ parameters in Equations (12.5.1-1), (12.5.2-1), (12.5.3-1) and subsequent Equation (12.5.4-1) remain valid as discussed in Section 12.5.

$$
\begin{aligned}
\dot{\underline{\gamma}}^N &= - C_B^N\, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\gamma}^N + \frac{1}{R}\, \underline{\omega}_{IE}^N \times \left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) + \left(\underline{\omega}_{IE}^N \times \underline{u}_{ZN}^N\right) \varepsilon_{ZN} + \delta\underline{\omega}_{EN}^N \\
&\quad + C_B^N \left(\underline{\omega}_{IB}^B \times \delta\underline{\alpha}_{Quant}\right) + \frac{1}{r_l}\, \underline{u}_{ZN}^N \times \left(C_B^N\, \delta\underline{\upsilon}_{Quant}\right) \\
\delta\underline{\omega}_{IB}^B &= \delta K_{Scal/Mis}\, \underline{\omega}_{IB}^B + \delta\underline{K}_{Bias} + \delta\underline{\omega}_{Rand} \\
\dot{\varepsilon}_{ZN} &= - \frac{1}{R} \left(\underline{\omega}_{EN_H}^N \cdot \delta\underline{R}_H^N\right) + \delta\rho_{ZN} \qquad\qquad (12.5.3\text{-}1) \\
\delta\dot{\underline{V}}^N &= C_B^N\, \delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\gamma}^N - \Delta\underline{a}_{SF}^N \times \left[\frac{1}{R}\left(\underline{u}_{ZN}^N \times \delta\underline{R}^N\right) + \varepsilon_{ZN}\, \underline{u}_{ZN}^N\right] \\
&\quad - \left(\underline{\omega}_{IE}^N + \underline{\omega}_{IN}^N\right) \times \delta\underline{V}^N + \delta\underline{g}_{Mdl}^N + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta R + C_2\, \delta h_{Prsr} - \delta e_{vc_3}\right] \underline{u}_{ZN}^N \\
&\quad - \left(\underline{a}_{SF}^N \times\right) C_B^N\, \delta\underline{\alpha}_{Quant} - \left[\left(C_B^N\, \underline{\omega}_{IB}^B + \underline{\omega}_{IE}^N\right) \times\right] C_B^N\, \delta\underline{\upsilon}_{Quant}
\end{aligned}
$$

$$
F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0
$$

$$
\begin{aligned}
\delta\underline{a}_{SF}^B &= \delta L_{Scal/Mis}\, \underline{a}_{SF}^B + \delta\underline{L}_{Bias} + \delta\underline{a}_{Rand} \\
\delta\dot{\underline{R}}^N &= \delta\underline{V}^N - \underline{\omega}_{EN}^N \times \delta\underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^N + C_B^N\, \delta\underline{\upsilon}_{Quant} \\
\delta\dot{e}_{vc_3} &= C_1 \left(\delta R - \delta h_{Prsr}\right)
\end{aligned}
$$

(Continued)

$$\delta\underline{\omega}_{EN}^{N} = \delta\rho_{ZN}\, \underline{u}_{ZN}^{N} + \frac{1}{r_l}\left(\underline{u}_{ZN}^{N} \times \delta\underline{V}^{N}\right) - v_{ZN}\frac{1}{r_l^2}\left(\underline{u}_{ZN}^{N} \times \delta\underline{R}^{N}\right)$$

$$+ \frac{1}{r_l}\,\underline{v}_{H}^{N}\,\varepsilon_{ZN} - \frac{1}{r_l^2}\left(\underline{u}_{ZN}^{N} \times \underline{v}^{N}\right)\delta R \qquad \begin{array}{c}(12.5.3\text{-}1)\\(\text{Continued})\end{array}$$

$\delta\rho_{ZN} = 0$ For Wander Azimuth Implementation

$\delta\rho_{ZN} = -\frac{1}{R}\,\underline{\omega}_{IE_H}^{N} \cdot \delta\underline{R}_{H}^{N}$ For Free Azimuth Implementation

$$\delta R = \underline{u}_{ZN}^{N} \cdot \delta\underline{R}^{N}$$

$$\delta\underline{R}_{H}^{N} = \delta\underline{R}^{N} - \delta R\, \underline{u}_{ZN}^{N}$$

$$\underline{\omega}_{EN_H}^{N} = \underline{\omega}_{EN}^{N} - \rho_{ZN}\, \underline{u}_{ZN}^{N}$$

$$\Delta\underline{a}_{SF}^{N} = \underline{a}_{SF}^{N} + \underline{g}_{P}^{N}$$

12.5.4 REVISED ERROR RATE EQUATIONS FOR I FRAME DEFINED ERROR PARAMETERS PROJECTED ONTO THE N FRAME

The navigation error equations defined in the I Frame utilize the $\underline{\psi}$, $\delta\underline{\upsilon}$, $\delta\underline{R}$ error parameters. Equations (12.3.7.2-6) summarize the rate equations for the $\underline{\psi}$, $\delta\underline{\upsilon}$, $\delta\underline{R}$ parameters projected in the N Frame. The equations for the strapdown inertial sensor errors in Equations (12.3.7.2-6) are given by Equations (12.4-12), (12.4-14) and (12.5-1). Following the development procedure used in Section 12.5 for the $\underline{\gamma}$, $\delta\underline{v}$, $\underline{\varepsilon}$, δh parameters, a revised set of error equations can be developed from Equations (12.3.7.2-6) in terms of the revised parameters $\underline{\psi}^{*N}$, $\delta\underline{\upsilon}^{*N}$, $\delta\underline{R}^{N}$ of which $\underline{\psi}^{*N}$, $\delta\underline{\upsilon}^{*N}$ are defined by Equations (12.5-21) and (12.5-23) as not containing sensor quantization noise. Following the generalization plan in Section 12.5, the revised $\underline{\psi}^{*N}$, $\delta\underline{\upsilon}^{*N}$, $\delta\underline{R}^{N}$ rate equations can then be simplified by deleting the asterisk * notation. Such a revised $\underline{\psi}$, $\delta\underline{\upsilon}$, $\delta\underline{R}$ equation set projected in the N Frame (revised from Equations (12.3.7.2-6)) is summarized below by Equations (12.5.4-1) together with remaining applicable elements of (12.3.7.2-6). The $\underline{\psi}$, $\delta\underline{\upsilon}$ parameters in Equations (12.5.4-1) then, are the $\underline{\psi}^{*N}$, $\delta\underline{\upsilon}^{*N}$ terms defined by Equations (12.5-21) and (12.5-23) with quantization noise removed, but with the asterisk * notation deleted for simplicity (not to be confused with the $\underline{\psi}$, $\delta\underline{\upsilon}$ parameters in Equations (12.3.7.2-6) which include quantization noise error). The equivalency relationships in Sections 12.2.1 and 12.2.2 between the $\underline{\gamma}$, $\underline{\psi}$, $\delta\underline{v}$, $\delta\underline{V}$, $\delta\underline{\upsilon}$ parameters in Equations (12.5.1-1), (12.5.2-1), (12.5.3-1) and (12.5.4-1) remain valid as discussed in Section 12.5.

$$
\begin{aligned}
\dot{\underline{\psi}}^N &= - C_B^N\, \delta\underline{\omega}_{IB}^B - \underline{\omega}_{IN}^N \times \underline{\psi}^N + C_B^N \left(\underline{\omega}_{IB}^B \times \delta\underline{\alpha}_{Quant}\right) \\
\delta\underline{\omega}_{IB}^B &= \delta K_{Scal/Mis}\, \underline{\omega}_{IB}^B + \delta\underline{K}_{Bias} + \delta\underline{\omega}_{Rand} \\
\delta\dot{\underline{\upsilon}}^N &= C_B^N\, \delta\underline{a}_{SF}^B + \underline{a}_{SF}^N \times \underline{\psi}^N - \frac{g}{R}\, \delta\underline{R}_H^N - \underline{\omega}_{IN}^N \times \delta\underline{\upsilon}^N + \delta\underline{g}_{Mdl}^N \\
&\quad + \left[\left(F(h)\, \frac{g}{R} - C_2\right) \delta R - \delta e_{vc_3} + C_2\, \delta h_{Prsr}\right] \underline{u}_{ZN}^N \\
&\quad - \left(\underline{a}_{SF}^N \times\right) C_B^N\, \delta\underline{\alpha}_{Quant} - \left[\left(C_B^N\, \underline{\omega}_{IB}^B\right) \times\right] C_B^N\, \delta\underline{\upsilon}_{Quant}
\end{aligned}
\qquad (12.5.4\text{-}1)
$$

$F(h) = 2 \quad \text{For} \quad h \geq 0 \qquad\qquad F(h) = -1 \quad \text{For} \quad h < 0$

$$
\begin{aligned}
\delta\underline{a}_{SF}^B &= \delta L_{Scal/Mis}\, \underline{a}_{SF}^B + \delta\underline{L}_{Bias} + \delta\underline{a}_{Rand} \\
\delta\dot{\underline{R}}^N &= \delta\underline{\upsilon}^N - \underline{\omega}_{IN}^N \times \delta\underline{R}^N - C_3 \left(\delta R - \delta h_{Prsr}\right) \underline{u}_{ZN}^N + C_B^N\, \delta\underline{\upsilon}_{Quant} \\
\delta\dot{e}_{vc_3} &= C_1 \left(\delta R - \delta h_{Prsr}\right) \\
\delta\underline{R}_H^N &= \delta\underline{R}^N - \delta R\, \underline{u}_{ZN}^N \\
\delta R &= \underline{u}_{ZN}^N \cdot \delta\underline{R}^N
\end{aligned}
$$

12.5.5 REVISED ERROR RATE EQUATIONS FOR I FRAME DEFINED ERROR PARAMETERS PROJECTED ONTO THE I FRAME

The equivalent to Equations (12.5.4-1) in the I Frame (i.e., for $\underline{\psi}^I$, $\delta\underline{\upsilon}^I$, $\delta\underline{R}^I$) is easily obtained by equating N to I which sets $\underline{\omega}_{IN}$ to zero. The result is what would have been achieved by converting Equations (12.3.7.1-15) directly to the * format and then deleting the * notation.

12.5.6 INERTIAL SENSOR ERROR RATE EQUATIONS

Error equation sets (12.5.1-1), (12.5.2-1), (12.5.3-1) and (12.5.4-1) have common general inertial sensor error models:

$$\begin{aligned} \delta\underline{\omega}_{IB}^{B} &= \delta K_{Scal/Mis}\, \underline{\omega}_{IB}^{B} + \delta\underline{K}_{Bias} + \delta\underline{\omega}_{Rand} \\ \delta\underline{a}_{SF}^{B} &= \delta L_{Scal/Mis}\, \underline{a}_{SF}^{B} + \delta\underline{L}_{Bias} + \delta\underline{a}_{Rand} \end{aligned} \qquad (12.5.6\text{-}1)$$

As discussed in Section 12.4, the $\delta\underline{\omega}_{Rand}$ and $\delta\underline{a}_{Rand}$ terms in (12.5.6-1) can be treated analytically as white noise. The inertial sensor quantization noise terms $\delta\underline{\alpha}_{Quant}$ and $\delta\underline{\upsilon}_{Quant}$ (that are individually included in navigation error Equations (12.5.1-1), (12.5.2-1), (12.5.3-1) and (12.5.4-1) as part of the navigation error parameter rate expressions) are also modeled as white noise. The remaining residual sensor error terms in Equations (12.5.6-1) ($\delta K_{Scal/Mis}$, $\delta\underline{K}_{Bias}$, $\delta L_{Scal/Mis}$ and $\delta\underline{L}_{Bias}$) are modeled by differential equations to reflect the principal error characteristics of the particular inertial sensors in the system following calibration/compensation in the system computer. For some sensors (e.g., ring laser gyros) the scale factor, misalignment and bias error residuals can be adequately modeled as unknown constants with a slow random buildup. For example, the dominant "constant" contribution to the angular rate sensor $\delta\underline{K}_{Bias}$ term might be modeled in differential equation format as:

$$\delta\dot{\underline{K}}_{0Bias} = \underline{n}_{K0Bias} \qquad (12.5.6\text{-}2)$$

with an unknown initial value for $\delta\underline{K}_{0Bias}$ and where:

$\delta\underline{K}_{0Bias}$ = Contribution to $\delta\underline{K}_{Bias}$ modeled as an unknown initial value that changes randomly from instant to instant.

$\underline{n}_{K0Bias}$ = White noise vector representing the rate of change of $\delta\underline{K}_{0Bias}$. Each component of $\underline{n}_{K0Bias}$ is uncorrelated from its other components.

The form of each component of the (12.5.6-2) vector equation is known as a random walk process due to the characteristic of the integral to wander randomly from instant to instant from its previous value.

The model for $\delta\underline{K}_{Bias}$ vector might also include what is known as a first order Markov process for each vector component which has the combined vector form:

$$\delta\dot{\underline{K}}_{MBias} = - C_{KMBias}\, \delta\underline{K}_{MBias} + \underline{n}_{KMBias} \qquad (12.5.6\text{-}3)$$

where

$\delta\underline{K}_{MBias}$ = Portion of $\delta\underline{K}_{Bias}$ characterized as a vector first order Markov process.

C_{KMBias} = $\delta \underline{K}_{MBias}$ "correlation frequency" coefficient. A more general formulation would have a distinct value of C_{KMBias} for each of the $\delta \underline{K}_{MBias}$ vector components (i.e., C_{KMBias} represented as a diagonal matrix with distinct elements on the diagonal).

$\underline{n}_{KMBias}$ = $\delta \underline{K}_{MBias}$ Markov process input white noise vector. Each component of $\underline{n}_{KMBias}$ is uncorrelated from its other components.

The distinction between the Equation (12.5.6-2) and (12.5.6-3) processes is that the Markov process reaches a steady state condition in which the statistical ensemble amplitude (i.e., the root-mean-square (RMS) value over an ensemble of samples) is constant (see Section 16.2.3.1 for analytical discussion); for the random walk process, the average squared ensemble amplitude value is unbounded, building linearly with time, hence, the RMS amplitude builds as the square root of time. The above property of the random walk process can be easily seen from (12.5.6-2) using generalized Equation (15.1-1) (compared with (12.5.6-2)), (15.1.2.1-5) and (15.1.2.1.1-30).

In general, the angular rate $\delta \underline{K}_{Bias}$ term can be characterized as the sum of random walk, Markov process and "other effects" that are unique to the particular sensor being analyzed. The "other effects" might include residual terms that are linearly proportional to specific force acceleration components (i.e., "g" sensitive errors) or proportional to products of specific force acceleration components (e.g., "g-squared" sensitive errors such as discussed in Section 12.6). The coefficients for the "other effects" would also be modeled as random walk processes plus (if significant) Markov processes. The random walk process associated with a particular sensor residual error coefficient is generally larger in magnitude (statistically) than the associated first order Markov process. This provides the rationale in some applications for dropping the Markov process terms as negligible. However, if Kalman filter aiding is incorporated in the system and is used to correct the residual error in a particular sensor error coefficient, the random walk term statistical amplitude might be reduced to a level at which the Markov process term becomes significant. A simplified version of random-walk/Markov process Equations (12.5.6-2) - (12.5.6-3) is also possible embodying both of the previous effects. With this approach, $\delta \underline{K}_{0Bias}$ and $\delta \underline{K}_{MBias}$ would be merged into a combined single error state that uses the $\delta \dot{\underline{K}}_{0Bias}$ model initially (when $\delta \underline{K}_{0Bias}$ is large) and then switches to the $\delta \dot{\underline{K}}_{MBias}$ model when the $\delta \underline{K}_{0Bias}$ variance is reduced to the $\delta \underline{K}_{MBias}$ initial uncertainty level by the Kalman filter estimation process. A similar discussion applies to the $\delta \underline{L}_{Bias}$ term for the accelerometer.

In the case of the $\delta K_{Scal/Mis}$ and $\delta L_{Scal/Mis}$ scale-factor/misalignment residual error terms, random walk processes are generally sufficient for analytical modeling, with Markov processes sometimes included for aided systems in which the particular error terms are being estimated and corrected by a Kalman filter. For inertial sensors having significant residual scale factor non-linearity errors (e.g., accelerometers with electronic output pulse generators), these effects may require additional random constant coefficients for modeling that are proportional to the associated particular sensor input rectification effect (e.g., input squared or absolute value of input). Regarding modeling of scale factor non-linearity error, it is important to consider the effect of sensor input vibration which may be the dominant rectification error produced (See Section 12.6 for example).

12.6 VIBRATION MODELING

Vibration effects (both angular and linear) appear as angular and linear vibration components superimposed on $\underline{\omega}_{IB}$ and $\underline{a}_{SF}$ in navigation error Equations (12.5.1-1), (12.5.2-1), (12.5.3-1) and (12.5.4-1), and in the inertial sensor analytical models for vibration sensitive error effects in these equations (as discussed in Section 12.5.6).

The following examples illustrate how vibration effects on X-axis inertial sensor models might be handled for a conventional spinning wheel gyro with "g-squared" sensitive error (as in Section 16.2.3.2) and for an accelerometer with scale factor asymmetry error:

$$\begin{aligned} \delta K_{G2Bias_X} &= \left(a_{Vib}^2 + a_{IA_X}\, a_{SA_X}\right) \delta\omega_{G2Bias_X} \\ \delta L_{SFAsymBias_X} &= \left(a_{Vib}^2 + a_{IA_X}^2\right) \delta a_{SFAsym_X} \end{aligned} \qquad (12.6\text{-}1)$$

where

a_{Vib} = Root-mean-square specific force acceleration vibration amplitude per axis.

a_{IA_X} = Component of $\underline{a}_{SF_{Mean}}^B$ (defined below) along the X-axis inertial sensor input axis.

a_{SA_X} = Component of $\underline{a}_{SF_{Mean}}^B$ (defined below) along the X-axis gyro spin axis.

$\underline{a}_{SF_{Mean}}^B$ = Mean value portion of $\underline{a}_{SF}^B$ (i.e., exclusive of vibration effects).

δK_{G2Bias_X} = Contribution to $\delta\underline{K}_{Bias}$ from the X-axis angular rate sensor caused by residual (after compensation) g-squared error sensitivity.

$\delta\omega_{G2Bias_X}$ = X-axis angular rate sensor residual g-squared bias error coefficient.

$\delta L_{SFAsymBias_X}$ = Contribution to $\delta\underline{L}_{Bias}$ from the X-axis accelerometer caused by residual (after compensation) asymmetrical scale factor error sensitivity.

δa_{SFAsym_X} = X-axis accelerometer residual scale factor asymmetry error coefficient.

The $\delta\omega_{G2Bias_X}$ and δa_{SFAsym_X} residual sensor error coefficients in Equations (12.6-1) would typically be modeled as unknown constants. Note, that for Kalman filter aided applications, these coefficients would not be estimated by the Kalman filter because the a_{Vib} term is typically not available in the system computer, and because the error model has some degree of uncertainty. It might, however, be included in "optimal" Kalman filter simulation analyses, but with the associated Kalman gains set to zero to account for the error but to not estimate it (i.e., the "considered variable" approach). See Section 15.1.2.1.1 for more detail.

The effects of vibration on $\underline{\omega}_{IB}$ and $\underline{a}_{SF}$ in navigation error Equations (12.5.1-1), (12.5.2-1), (12.5.3-1) and (12.5.4-1) primarily affect the product of these terms with the inertial sensor quantization noise $\delta\underline{\alpha}_{Quant}$ and $\delta\underline{\upsilon}_{Quant}$. A method that can be used for modeling this effect is to treat $\underline{\omega}_{IB}$ and $\underline{a}_{SF}$ multiplying $\delta\underline{\alpha}_{Quant}$ and $\delta\underline{\upsilon}_{Quant}$ (which only occurs in the attitude and velocity error rate equations) as the sum of vibration and "mean" effects, with the "mean" effects identified using the same $\underline{\omega}_{IB}$ and $\underline{a}_{SF}$ nomenclature. The remaining $\underline{\omega}_{IB}$ and $\underline{a}_{SF}$ terms in the navigation error equations would then be interpreted as representing the "mean" angular rate and linear acceleration with vibration excluded. With this approximation, the sensor quantization error effects in the Equations (12.5.1-1), (12.5.2-1), (12.5.3-1) and (12.5.4-1) attitude and velocity error rate expressions would be replaced by:

For $\dot{\underline{\psi}}^N$ in Equations (12.5.1-1):

$$C_B^N \left(\underline{\omega}_{IB}^B \times \delta\underline{\alpha}_{Quant}\right) \rightarrow C_B^N \left(\underline{\omega}_{IB}^B \times \delta\underline{\alpha}_{Quant}\right) + C_B^N \left(\underline{\omega}_{Vib}^B \times \delta\underline{\alpha}_{Quant}\right) \tag{12.6-2}$$

For $\delta\dot{\underline{V}}^N$ in Equations (12.5.1-1):

$$\begin{aligned} &- \left(\underline{a}_{SF}^N \times\right) C_B^N \, \delta\underline{\alpha}_{Quant} - \left[\left(C_B^N \, \underline{\omega}_{IB}^B\right) \times\right] C_B^N \, \delta\underline{\upsilon}_{Quant} \rightarrow \\ &\quad - \left(\underline{a}_{SF}^N \times\right) C_B^N \, \delta\underline{\alpha}_{Quant} - \left[\left(C_B^N \, \underline{\omega}_{IB}^B\right) \times\right] C_B^N \, \delta\underline{\upsilon}_{Quant} \\ &\qquad - C_B^N \left(\underline{a}_{Vib}^B \times \delta\underline{\alpha}_{Quant} + \underline{\omega}_{Vib}^B \times \delta\underline{\upsilon}_{Quant}\right) \end{aligned} \tag{12.6-3}$$

For $\dot{\underline{\gamma}}^N$ in Equations (12.5.2-1):

$$C_B^N \left(\underline{\omega}_{IB}^B \times \delta\underline{\alpha}_{Quant}\right) \rightarrow C_B^N \left(\underline{\omega}_{IB}^B \times \delta\underline{\alpha}_{Quant}\right) + C_B^N \left(\underline{\omega}_{Vib}^B \times \delta\underline{\alpha}_{Quant}\right) \tag{12.6-4}$$

For $\delta \underline{\dot{v}}^N$ in Equations (12.5.2-1):

$$- \left(\underline{a}_{SF}^N \times \right) C_B^N \, \delta \underline{\alpha}_{Quant} - \left[\left(C_B^N \underline{\omega}_{IB}^B \right) \times \right] C_B^N \, \delta \underline{\upsilon}_{Quant} \;\rightarrow$$
$$- \left(\underline{a}_{SF}^N \times \right) C_B^N \, \delta \underline{\alpha}_{Quant} - \left[\left(C_B^N \underline{\omega}_{IB}^B \right) \times \right] C_B^N \, \delta \underline{\upsilon}_{Quant}$$
$$- C_B^N \left(\underline{a}_{Vib}^B \times \delta \underline{\alpha}_{Quant} + \underline{\omega}_{Vib}^B \times \delta \underline{\upsilon}_{Quant} \right) \tag{12.6-5}$$

For $\underline{\dot{\gamma}}^N$ in Equations (12.5.3-1):

$$C_B^N \left(\underline{\omega}_{IB}^B \times \delta \underline{\alpha}_{Quant} \right) \;\rightarrow\; C_B^N \left(\underline{\omega}_{IB}^B \times \delta \underline{\alpha}_{Quant} \right) + C_B^N \left(\underline{\omega}_{Vib}^B \times \delta \underline{\alpha}_{Quant} \right) \tag{12.6-6}$$

For $\delta \underline{\dot{V}}^N$ in Equations (12.5.3-1):

$$- \left(\underline{a}_{SF}^N \times \right) C_B^N \, \delta \underline{\alpha}_{Quant} - \left[\left(C_B^N \underline{\omega}_{IB}^B \right) \times \right] C_B^N \, \delta \underline{\upsilon}_{Quant} \;\rightarrow$$
$$- \left(\underline{a}_{SF}^N \times \right) C_B^N \, \delta \underline{\alpha}_{Quant} - \left[\left(C_B^N \underline{\omega}_{IB}^B \right) \times \right] C_B^N \, \delta \underline{\upsilon}_{Quant}$$
$$- C_B^N \left(\underline{a}_{Vib}^B \times \delta \underline{\alpha}_{Quant} + \underline{\omega}_{Vib}^B \times \delta \underline{\upsilon}_{Quant} \right) \tag{12.6-7}$$

For $\underline{\dot{\psi}}^N$ in Equations (12.5.4-1):

$$C_B^N \left(\underline{\omega}_{IB}^B \times \delta \underline{\alpha}_{Quant} \right) \;\rightarrow\; C_B^N \left(\underline{\omega}_{IB}^B \times \delta \underline{\alpha}_{Quant} \right) + C_B^N \left(\underline{\omega}_{Vib}^B \times \delta \underline{\alpha}_{Quant} \right) \tag{12.6-8}$$

For $\delta \underline{\dot{\upsilon}}^N$ in Equations (12.5.4-1):

$$- \left(\underline{a}_{SF}^N \times \right) C_B^N \, \delta \underline{\alpha}_{Quant} - \left[\left(C_B^N \underline{\omega}_{IB}^B \right) \times \right] C_B^N \, \delta \underline{\upsilon}_{Quant} \;\rightarrow$$
$$- \left(\underline{a}_{SF}^N \times \right) C_B^N \, \delta \underline{\alpha}_{Quant} - \left[\left(C_B^N \underline{\omega}_{IB}^B \right) \times \right] C_B^N \, \delta \underline{\upsilon}_{Quant}$$
$$- C_B^N \left(\underline{a}_{Vib}^B \times \delta \underline{\alpha}_{Quant} + \underline{\omega}_{Vib}^B \times \delta \underline{\upsilon}_{Quant} \right) \tag{12.6-9}$$

where

$\underline{\omega}_{Vib}^B$ = Angular rate vibration of B frame relative to inertial space as projected on B Frame axes.

$\underline{a}^B_{Vib}$ = Specific force acceleration vibration of the B Frame relative to inertial space as projected on B Frame axes.

Equations (12.6-8) - (12.6-9) are also compatible with the I Frame version of Equations (12.5.4-1) if we replace N with I. The equivalent to (12.5.4-1) (for $\underline{\psi}^I$, $\delta\underline{\upsilon}^I$, $\delta\underline{R}^I$) is obtained by equating N to I which sets $\underline{\omega}_{IN}$ to zero.

The Equation (12.6-1) - (12.6-11) vibration and noise effects (as well as the noise effects in the overall navigation equation sets (12.5.1-1), (12.5.2-1), (12.5.3-1) and (12.5.4-1)) are typically analyzed using covariance simulation analysis techniques as discussed in Chapter 16.

References

1. Bisplinghoff, R. L., Ashley, H. & Halfman, R. L., *Aeroelasticity*, Addison-Wesley, Reading Mass., 1956.

2. Bortz J. E., “A New Mathematical Formulation for Strapdown Inertial Navigation”, *IEEE Transactions on Aerospace and Electronic Systems*, Volume AES-7, No. 1, January 1971, pp. 61-66.

3. Britting, K. R., *Inertial Navigation System Analysis*, John Wiley and Sons, New York, 1971.

4. “Department Of Defense World Geodetic System 1984”, NIMA TR8350.2, Third Edition, 4 July 1997.

5. Downs, H. B., “A Lab Test To Find The Major Error Sources In A Laser Strapdown Inertial Navigator”, 38th Annual Meeting of the ION, Colorado Springs, CO, June 15-17, 1982.

6. Gelb, A., *Applied Optimal Estimation*, The MIT Press, Cambridge Mass., London, England, 1978.

7. Gille, J. C., Pelegrin, M. J., & Decaulne, P., *Feedback Control Systems Analysis, Synthesis, And Design*, McGraw-Hill, New York, Toronto, London, 1959.

7a. Goodman, L.E. and Robinson, A.R., "Effects of Finite Rotations on Gyroscope Sensing Devices", *Journal of Applied Mechanics*, Vol. 25, June 1958.

8. Halfman, R. L., *Dynamics: Particles, Rigid Bodies, and Systems, Volume I*, Addison-Wesley, Reading Mass., Palo Alto, London, 1962.

9. Hills, F. B., “A Study Of Coordinate-Conversion Errors In Strapped-Down Navigation”, MIT Electronics Systems Laboratory, E SL-4-244, Cambridge, MA, August 1965.

10. Holbrook, J. G., *Laplace Transforms For Electronics Engineers*, Pergamon Press, New York, London, Paris, Los Angeles, 1959.

11. Ignagni, M. B., “Optimal Strapdown Attitude Integration Algorithms”, *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 13, No. 2, March-April 1990, pp. 363-369.

11a. Ignagni, M.B., "On the Orientation Vector Differential Equation in Strapdown Inertial Systems", *IEEE Transactions On Aerospace and Electronic Systems*, Vol. 30, No. 4, October 1994, pp. 1076-1081.

12. Ignagni, M. B., “Efficient Class Of Optimized Coning Compensation Algorithms”, *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 19, No. 2, March-April 1996, pp. 424-429.

13. Ignagni, M. B., “Duality of Optimal Strapdown Sculling and Coning Compensation Algorithms”, *Journal of the ION*, Vol. 45, No. 2, Summer 1998.

14. Jordan, J. W., “An Accurate Strapdown Direction Cosine Algorithm”, NASA TN-D-5384, Sept.ember 1969.

15. Kachickas, G. A., “Error Analysis For Cruise Systems”, *Inertial Guidance*, edited by Pitman, G. R., Jr., John Wiley & Sons, New York, London, 1962.

15a. Laning, J.H., Jr., "The Vector Analysis of Finite Rotations and Angles", Massachusetts Institute of Technology, Cambridge, Instrumentation Laboratory Special Report 6398-S-3, 1949.

16. Lawrence, Anthony, *Modern Inertial Technology*, Springer-Verlag New York, Inc., 1993.

17. Liepmann, H. W. & Roschko, A., *Elements of Gasdynamics*, John Wiley & Sons, New York, and Chapman & Hall, London, 1957.

18. Litmanovich, Y. A., Lesyuchevsky, V. M. & Gusinsky, V. Z., “Two New Classes of Strapdown Navigation Algorithms”, *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 23, No. 1, January- February 2000.

19. Mark, J.G. & Tazartes, D.A., “On Sculling Algorithms”, 3rd St. Petersburg International Conference On Integrated Navigation Systems, St. Petersburg, Russia, May 1996.

19a. Martin, William Ted and Reissner, Eric, *Elementary Differential Equations*, Addison-Wesley, Cambridge Mass., 1956, Sec. 6-8.

20. Mckern, R. A., “A Study of Transformation Algorithms For Use In A Digital Computer”, Massachusetts Institute of Technology, Master’s Thesis, Department of Aeronautics and Astronautics, Cambridge, MA, January 1968.

21. Merhav, Schmuel, *Aerospace Sensor Systems and Applications*, Springer-Verlag New York, Inc., 1996.

21a. Meriam, J. L., *Mechanics Part II . Dynamics*, John Wiley & Sons, 1951, 1952, pp. 664

22. Miller, R., “A New Strapdown Attitude Algorithm”, *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 6, No. 4, July-August 1983, pp. 287-291.

23. Minor, J. W., “Low-Cost Strapdown-Down Inertial Systems”, AIAA/ION Guidance and Control Conference, August 16-18, 1965.

24. Morrisson, N., *Introduction to Sequential Smoothing and Prediction*, Mcgraw-Hill, New York, St. Louis, San Francisco, London, Sydney, Toronto, Mexico, Panama, 1969.

25. Morse, P. M. and Feshbach, H., *Methods of Theoretical Physics, Part 1*, McGraw-Hill, New York, Toronto, London, 1953.

26. Newton, G. C., Gould, L. A. & Kaiser, J. F., *Analytical Design of Linear Feedback Controls*, John Wiley & Sons, New York, and Chapman & Hall, London, 1957.

27. Parkinson, B. W. & Spilker, Jr., J. J., *Global Positioning System: Theory and Applications, Volume I*, American Institute of Aeronautics & Astronautics, Washington DC, 1996.

28. Perkins, C. D. & Hage, R. E., *Airplane Performance Stability and Control*, John Wiley & Sons, New York, and Chapman & Hall, London, 1957.

28a. Roscoe, K. M., "Equivalency Between Strapdown Inertial Navigation Coning and Sculling Integrals/Algorithms", *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 24, No. 2, March-April 2001, pp. 201-205.

29. Savage, P. G., "A New Second-Order Solution for Strapped-Down Attitude Computation", AIAA/JACC Guidance & Control Conference, Seattle, Washington, August 15-17, 1966.

30. Savage, P. G., "Calibration Procedures For Laser Gyro Strapdown Inertial Navigation Systems", 9th Annual Electro-Optics / Laser Conference and Exhibition, Anaheim, California, October 25 - 27, 1977.

31. Savage, P. G., "Strapdown Sensors", *Strapdown Inertial Systems - Theory And Applications*, NATO AGARD Lecture Series No. 95, June 1978, Section 2.

32. Savage, P. G., "Advances In Strapdown Sensors", *Advances In Strapdown Inertial Systems*, NATO AGARD Lecture Series No. 133, May 1984, Section 2.

33. Savage, P. G., "Strapdown System Algorithms", *Advances In Strapdown Inertial Systems*, NATO AGARD Lecture Series No. 133, May 1984, Section 3.

34. Savage, P. G., "Strapdown Inertial Navigation System Integration Algorithm Design Part 1 - Attitude Algorithms", *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 21, No. 1, January-February 1998, pp. 19-28.

35. Savage, P. G., "Strapdown Inertial Navigation System Integration Algorithm Design Part 2 - Velocity and Position Algorithms", *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 21, No. 2, March-April 1998, pp. 208-221.

35a. Savage, P. G., "Analytical Modeling of Sensor Quantization in Strapdown Inertial Navigation Error Equations", *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 25, No. 5, September-October 2002, pp. 833-842.

35b. Savage, P. G., "A Unified Mathematical Framework For Strapdown Algorithm Design", *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 29, No. 2, March-April 2006, pp. 237-249.

35c. Savage, P. G., "Reply by the Author to Y. Wu", *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 29, No. 6, November-December 2006, pp. 1485-1486.

36. Shepperd, S. W., “Quaternion From Rotation Matrix”, *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 1, No. 3, May- June 1978.

36a. Shuster, Malcolm D., "The Kinematic Equation for the Rotation Vector", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, Issue 1, January 1993, pp. 263-267

37. Sikolnikoff, I. S. & Redheffer, R. M., *Mathematics of Physics and Modern Engineering, McGraw-Hill*, New York, Toronto, London, 1958.

38. Thomas, Jr., G. B., *Calculus And Analytic Geometry*, Addison-Wesley, Cambridge Mass., 1955.

39. Turley, A. R., “A Solution For The Problems Of The No-Gimbal Inertial Navigator Concept”, Air Force Avionics Laboratory, AFAL-TR-64-307, Wright Patterson AFB, OH, January 1965.

40. United Aircraft Corporation, “A Study of Critical Computational Problems Associated with Strapdown Inertial Navigation Systems”, NASA Report CR-968, April 1968.

40a. Wu, Y. "Comment on 'A Unified Mathermatical Framework for Strapdown Algorithm Design' ", *AIAA Journal Of Guidance, Control, And Dynamics*, Vol. 29, No. 6, November-December 2006, pp. 1482-1484.

Subject Index

(Subject Located By Section Number Unless Otherwise Indicated)

Coordinate Frame Index

Parameter Index

NOTES