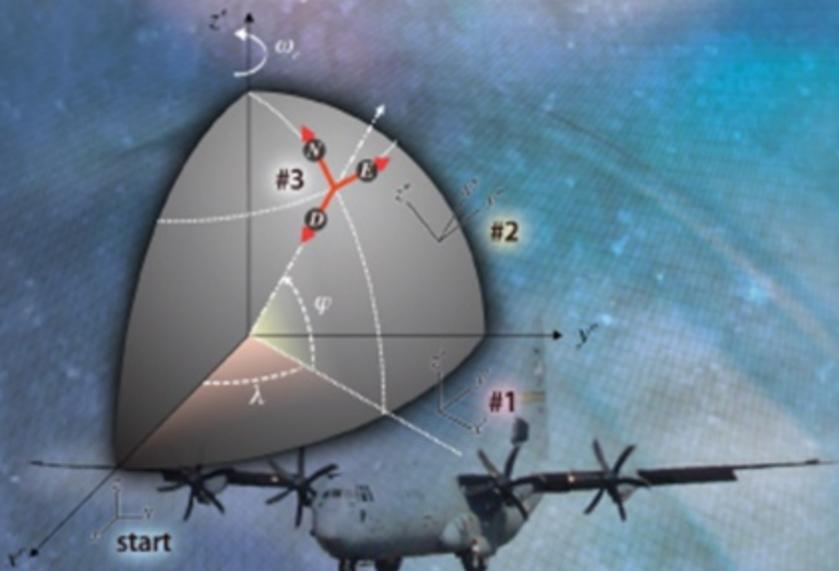


# APPLIED MATHEMATICS IN INTEGRATED NAVIGATION SYSTEMS

## THIRD EDITION

Robert M. Rogers



AMERICAN INSTITUTE OF  
AERONAUTICS AND ASTRONAUTICS

AIAA EDUCATION SERIES  
JOSEPH A. SCHETZ  
EDITOR-IN-CHIEF

# **Applied Mathematics in Integrated Navigation Systems**

**Third Edition**



# **Applied Mathematics in Integrated Navigation Systems**

## **Third Edition**

**Robert M. Rogers**

Rogers Engineering & Associates  
Gainesville, Florida



### **EDUCATION SERIES**

Joseph A. Schetz

Series Editor-in-chief

Virginia Polytechnic Institute and State University  
Blacksburg, Virginia

Published by

American Institute of Aeronautics and Astronautics, Inc.  
1801 Alexander Bell Drive, Reston, VA 20191-4344

American Institute of Aeronautics and Astronautics, Inc., Reston, Virginia

1 2 3 4 5

**Library of Congress Cataloging-in-Publication data on file**

ISBN-13: 978-1-56347-927-4

ISBN-10: 1-56347-927-3

Copyright © 2007 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.  
Printed in the United States of America. No part of this publication may be reproduced, distributed, or  
transmitted, in any form or by any means, or stored in a database or retrieval system, without the prior  
written permission of the publisher.

Data and information appearing in this book are for informational purposes only. AIAA is not responsible  
for any injury or damage resulting from use or reliance, nor does AIAA warrant that use or  
reliance will be free from privately owned rights.

## AIAA Education Series

### Editor-in-Chief

Joseph A. Schetz

*Virginia Polytechnic Institute and State University*

### Editorial Board

Takahira Aoki  
*University of Tokyo*

Rakesh K. Kapania  
*Virginia Polytechnic Institute  
and State University*

Edward W. Ashford

Brian Landrum  
*University of Alabama  
Huntsville*

Karen D. Barker  
*Brahe Corporation*

Robert H. Bishop  
*University of Texas at Austin*

Timothy C. Lieuwen  
*Georgia Institute of Technology*

Claudio Bruno  
*University of Rome*

Michael Mohaghegh  
*The Boeing Company*

Aaron R. Byerley  
*U.S. Air Force Academy*

Conrad F. Newberry  
*Naval Postgraduate School*

Richard Colgren  
*University of Kansas*

Mark A. Price  
*Queen's University Belfast*

Kajal K. Gupta  
*NASA Dryden Flight Research  
Center*

James M. Rankin  
*Ohio University*

Rikard B. Heslehurst  
*Australian Defence Force  
Academy*

David K. Schmidt  
*University of Colorado  
Colorado Springs*

David K. Holger  
*Iowa State University*

David M. Van Wie  
*Johns Hopkins University*



## **Foreword**

We are very happy to present the third edition of *Applied Mathematics in Integrated Navigation Systems* by Robert M. Rogers. The first two editions have been very well received, and we are certain that this comprehensive and in-depth treatment of such a timely and important topic in the aerospace field, as well as others, will be equally well received by the technical community. The book has sixteen chapters divided into two main parts and several appendices, including a bibliography, all in more than 400 pages. There is a detailed Preface and software for download.

Robert Rogers is extremely well qualified to write this book, because of his broad and deep expertise in the area. His command of the material is excellent, and he is able to organize and present it in a very clear manner.

The AIAA Education Book Series aims to cover a very broad range of topics in the general aerospace field, including basic theory, applications and design. Information about a complete list of titles can be found on the last page of this volume. The philosophy of the series is to develop textbooks that can be used in a university setting, instructional materials for continuing education and professional development courses, and also books that can serve as the basis for independent study. Suggestions for new topics or authors are always welcome.

**Joseph A. Schetz**

Editor-in-Chief

AIAA Education Series



# Table of Contents

<b>Preface . . . . .</b>	<b>xiii</b>
<b>Part 1. Elements of Integrated Navigation Systems</b>	
<b>1. Introduction . . . . .</b>	<b>3</b>
<b>2. Mathematical Preliminaries . . . . .</b>	<b>7</b>
2.1    Vector/Matrix Algebra . . . . .	7
2.2    Vector/Matrix Calculus . . . . .	13
2.3    Linearization Techniques . . . . .	17
2.4    Direction Cosine Matrices . . . . .	19
2.5    Miscellaneous Mathematical Topics . . . . .	30
2.6    Chapter Summary . . . . .	32
Problems . . . . .	32
<b>3. Coordinate Systems and Transformations . . . . .</b>	<b>49</b>
3.1    Coordinate Systems . . . . .	49
3.2    Coordinate Frame Transformations . . . . .	54
3.3    Chapter Summary . . . . .	64
Problems . . . . .	65
<b>4. Earth Models . . . . .</b>	<b>71</b>
4.1    Ellipsoid Geometry . . . . .	71
4.2    Ellipsoid Gravity . . . . .	79
4.3    Chapter Summary . . . . .	87
Problems . . . . .	87
<b>5. Terrestrial Navigation . . . . .</b>	<b>101</b>
5.1    Strap-Down Navigation Systems . . . . .	101
5.2    Local-Level Navigation-Frame Mechanization Equations . . . . .	102
5.3    Perturbation Form of Navigation System Error Equations . . . . .	105

5.4	Navigation System Attitude Error Equations: $\Psi$ Formulation . . . . .	113
5.5	Navigation System Error Equations Using Alternative Velocity Error . . . . .	113
5.6	Vertical Channel . . . . .	116
5.7	Chapter Summary . . . . . Problems . . . . .	119 120
<b>6.</b>	<b>Navigation Sensor Models</b> . . . . .	<b>135</b>
6.1	Gyro Performance Characterizations . . . . .	135
6.2	Sensor Error Models . . . . .	140
6.3	Chapter Summary . . . . . Problems . . . . .	149 149
<b>7.</b>	<b>Navigation Aids</b> . . . . .	<b>153</b>
7.1	Doppler Velocity Sensors . . . . .	154
7.2	Tactical-Air-Navigation Range . . . . .	157
7.3	Global-Positioning-System Range . . . . .	160
7.4	Forward-Looking Infrared Line-of-Sight Systems . . . . .	169
7.5	Chapter Summary . . . . . Problems . . . . .	172 172
<b>8.</b>	<b>Kalman Filtering</b> . . . . .	<b>181</b>
8.1	Recursive Weighted Least Squares: Constant Systems . . . . .	182
8.2	Recursive Weighted Least Squares: Dynamic Systems . . . . .	187
8.3	Discrete Linear Minimum Variance Estimator . . . . .	191
8.4	$U-D$ Factored Form . . . . .	194
8.5	Summed Measurements . . . . .	201
8.6	Combined Estimate from Two (or More) Kalman Filters . . . . .	204
8.7	Derivative-Free Estimation . . . . .	207
8.8	Chapter Summary . . . . . Problems . . . . .	221 222
<b>Part 2. Applications</b>		
<b>9.</b>	<b>Strap-Down Inertial Sensor Laboratory Calibration</b> . . . . .	<b>237</b>
9.1	Navigation Mechanization Review . . . . .	238
9.2	Sensor Error Model . . . . .	238
9.3	Solutions for Sensor Errors . . . . .	239
9.4	Data-Collection Rotation Sequences . . . . .	239
9.5	Observation Equations . . . . .	241
9.6	Processing Sequences . . . . .	244
9.7	Simulated Laboratory Data Calibration . . . . .	247
9.8	Chapter Summary . . . . .	250

<b>10. Flight-Test Evaluations</b>	<b>251</b>
10.1    Optical Tracking Trajectory Reconstruction	252
10.2    Tactical-Air-Navigation/Inertial-Navigation-Unit Reconstruction	256
10.3    Vehicle Dynamics with Radar Tracking Trajectory Reconstruction	260
10.4    Chapter Summary	273
<b>11. Inertial Navigation System Ground Alignment</b>	<b>275</b>
11.1    Initial Coarse Alignment and Resulting Errors	276
11.2    Fine-Alignment Kalman Filter	279
11.3    Simulated Ground Fine Alignment	281
11.4    Chapter Summary	287
<b>12. Integration via Kalman Filtering: Global-Positioning-System Receiver</b>	<b>289</b>
12.1    Global-Positioning-System Receiver Kalman Filter Configurations	290
12.2    Inertial-Navigation-System Configuration Kalman Filter	290
12.3    Simulated Global-Positioning-System-Receiver Inertial-Navigation-System Kalman Filter Operation	299
12.4    Chapter Summary	304
<b>13. In-Motion Alignment</b>	<b>305</b>
13.1    Transfer Alignment	306
13.2    Alignment Without Benefit of Attitude Initialization	315
13.3    Chapter Summary	324
<b>14. Integrated Differential Global Positioning System/Dead-Reckoning Navigation</b>	<b>327</b>
14.1    Dead-Reckoning Navigation Equations	328
14.2    Dead-Reckoning System Error Model	329
14.3    Differential Global-Positioning-System Position Observations	332
14.4    Integrated Dead-Reckoning/Differential Global-Positioning-System Implementation	332
14.5    Test Conditions	333
14.6    Test Results	334
14.7    Chapter Summary	337
<b>15. Attitude-Determination and Estimation</b>	<b>339</b>
15.1    Terrestrial Attitude-Determination	339
15.2    Attitude-Determination by Iteration	344
15.3    Attitude Estimation	347
15.4    Chapter Summary	359

<b>16. Summary</b>	<b>361</b>
<b>A. Pinson Error Model</b>	<b>365</b>
A.1 Navigation System Error Model	365
A.2 Global-Positioning-System Latitude and Longitude Position as Measurements	374
<b>B. Orbital Dynamics</b>	<b>377</b>
B.1 Orbital Position in Earth-Centered Frame	377
B.2 Orbital Velocity in Earth-Centered Frame	384
<b>C. Coarse-Alignment Error Equations</b>	<b>387</b>
C.1 Position Error Equations	387
C.2 Position Error Dynamics	388
C.3 Attitude Error Equations	390
C.4 Velocity Error Equations	392
<b>D. Fine-Alignment Error Equations</b>	<b>393</b>
<b>E. References</b>	<b>395</b>
<b>F. Bibliography</b>	<b>399</b>
F.1 Inertial Sensors	399
F.2 Kalman Filtering	399
F.3 Navigation Systems	400
F.4 Orbital Mechanics	400
<b>Index</b>	<b>401</b>
<b>Supporting Materials</b>	<b>409</b>

Software download information can be found at the end of the book on the Supporting Materials page.

## Preface

The subject of integrated navigation systems covered in this book is designed for those directly involved with the design, integration, and test and evaluation of navigation systems. It is assumed that the reader has a background in mathematics, including calculus. Integrated navigation systems is the combination of an onboard navigation solution (position, velocity, and attitude) and independent navigation data (aids to navigation) to update or correct navigation solutions. In this book, this combination is accomplished with Kalman filter algorithms.

This presentation is segmented into two parts. In the first part, elements of basic mathematics, kinematics, equations describing navigation systems/sensors and their error models, aids-to-navigation, and Kalman filtering are developed. Detailed derivations are presented and examples are given to aid in the understanding of these elements of integrated navigation systems. Problems are included to expand the application of the materials presented.

This edition includes software for selected Chapter, Section and Exercise material in a companion CD to enhance the learning experience of the reader. The included software has been developed using MATLAB/Simulink<sup>TM</sup> version 6.5 by The MathWorks, Inc.

### **Part 1: Elements of Integrated Navigation Systems**

The concept of integrated navigation systems is introduced in Chapter 1 by illustrating different elements of operational navigation systems. The majority of the material presented addresses navigation systems that are mechanized with inertial sensors, accelerometers, and gyros to form an inertial navigation system to provide navigation state data: position, velocity, attitude, etc. To remove errors resulting from initialization and sensor errors, the navigation system's state data are combined with other independent data from aids-to-navigation. This combination is accomplished via a Kalman filter algorithm. This algorithm implements mathematical models relating a navigation system's state errors to corresponding errors in aids-to-navigation data. Aids-to-navigation data can include position, velocity, line-of-sight angles, etc., from various onboard sources, e.g., global positioning system (GPS) receivers, Doppler velocity sensors, optical sensors, radar, etc. Relating navigation states to aids-to-navigation data via a mathematical model is the principal

problem in integrated navigation systems and is dealt with extensively in Part 1 of this book.

Mathematical preliminaries are presented in Chapter 2 to familiarize the reader with notation and operations with vectors, matrices, and their calculus. This review provides the basis for later developments of navigation and Kalman filter equations. Examples applying the material presented include the formation of the least-squares estimation problem. This formulation is used repeatedly in the development of Kalman filter equations as presented in this book. Linearization techniques for nonlinear equations are presented and are applied later to nonlinear navigation equations to obtain linearized forms for Kalman filter implementations. Direction cosine matrices are used to transform vector components among various reference coordinate frames. The formation and dynamics of these matrices are developed.

Various coordinate systems in current use, and transformations between them, are illustrated in Chapter 3. These coordinate systems are used to reference navigation states: position, velocity, and attitude, determined by the navigation solution and used for relating information from aids to navigation to navigation states' reference frames. Coordinate systems that are currently used for navigation systems are presented in detail, including Earth-centered, Earth-fixed (ECEF), local geodetic, wander azimuth, and line-of-sight. Examples of position and rate errors for various wander azimuth coordinate frame definitions are developed.

Geometrical shape and gravitational models for representing the Earth are presented in Chapter 4 to account for the Earth's oblate shape and mass distribution. Relationships between ECEF position  $x$ - $y$ - $z$  components and local-level latitude, longitude, and altitude positions are developed. A vehicle's position change in geographical coordinates is related to the local Earth relative velocity and Earth curvature. A general development of Earth gravitational potential and acceleration is presented and along with illustrations of simplifications for use in navigation.

Equations describing terrestrial navigation are developed in Chapter 5. Many linearized forms of these equations are developed, including perturbation and an alternative velocity error representation that yields the "computer frame" form. Simplified forms of these error equations used in Kalman filter implementations are presented. The inertial navigation system's vertical axis control and stabilization, assuming a barometric pressure reference, is examined, and the corresponding error equations for a Kalman filter implementation presented. The historically significant Pinson error model is reexamined.

Functional characterizations of inertial sensors used in navigation systems are developed in Chapter 6. Performance characterizations for ring-laser gyro and fiber-optic gyro sensors are developed. Generalized error models for gyros and accelerometers are presented that represent these sensors' fixed errors, e.g., biases, scale factor, and misalignments. Time-varying dynamic error models for sensor random errors, useful for numerical simulations, are presented based on continuous system and stochastic differential equation approaches.

Functional characterizations of aids to navigation are presented in Chapter 7. These aids provide a source of independent navigation information that can be related to navigation states. These data are used in a navigation Kalman filter implementation to estimate errors associated with the navigation system.

Navigation aids presented include Doppler velocity sensors, tactical air navigation (TACAN) ranging, GPS-satellite ranging, and line-of-sight sensors. Examples relating the information from these aids in their respective reference frames to that used by the navigation system's Kalman filter are presented.

In Chapter 8, Kalman filtering equations are developed based on a recursive weighted least-squares approach. The Kalman filter algorithm is used to estimate navigation system errors by combining information available from the aids to navigation with navigation solution system data to obtain an optimal estimate of the navigation system's errors. Various forms of the Kalman filter algorithm are presented, including the conventional, Joseph's, and  $U-D$  factored forms. Kalman filtering techniques are extended to include modifications to the algorithm for including summed measurements and combining outputs of several independent parallel filters to form a single optimal estimate. Finally, a filtering algorithmic approaches that are derivative free, the Unscented Kalman Filter (UKF) and the Divided Difference Kalman Filter (DDKF), are presented.

## Part 2: Applications

Presented in the second part are case studies illustrating the application of the elements in Part 1 to integrated navigation systems.

In Chapter 9, the first of these case studies addresses the inertial navigation sensors' laboratory or factory calibration. In this case study, navigation error dynamic equations are treated as algebraic equations and a tracking model implemented in a Kalman filter is used to reconstruct higher derivative error states. With these higher derivative states available, the fixed sensor errors are estimated from a series of rotations of an accelerometer/gyro cluster assembly. Results are presented from simulated inertial sensor data, allowing a direct evaluation of calibration accuracies.

Trajectory reconstruction is used during flight test and evaluation of inertial navigation systems to provide an improved source of independent navigation state data by which to gauge the performance of the navigation system under test. Three trajectory reconstruction approaches are presented in Chapter 10. A least-squares technique using ground-based, line-of-sight tracking; a Kalman filter implementation using an onboard inertial navigation system aided with TACAN ranging; and a Kalman filter modeling a sounding rocket's dynamics integrated with ground-based radar track data are presented. Data from actual flight tests are used to illustrate the results obtained from these approaches.

Autonomous inertial navigation system ground alignment is illustrated in Chapter 11. Initial coarse alignment based on accelerometer and gyro sensor outputs is presented, with a characterization of the relationship between sensor error and the resulting alignment accuracy. An illustration of a final fine alignment using Kalman filtering is also presented. Simulated data are used to illustrate the accuracy of alignments achieved.

In Chapter 12, an operational GPS receiver unit integrated with an inertial navigation unit is the first integrated navigation system case study. In this case study, the GPS receiver's internal Kalman filter is used to estimate errors associated with the navigation state data being provided to the unit. The GPS receiver

and inertial navigation unit are loosely coupled. The Kalman filter implementation used by the GPS receiver is less complete in that it does not incorporate inertial sensor error states within its state vector. Outputs from the receiver's Kalman filter are combined with the inertial navigation unit's uncorrected outputs to form an optimal estimate of position, velocity, etc. Simulated data are used to illustrate the accuracy of the receiver's Kalman filter estimates.

In-motion alignment of an inertial measurement unit is presented in Chapter 13. This case study, which also includes establishing another navigation solution form, is presented for two major subcases. The first is a conventional transfer alignment with attitude initialization provided by another inertial navigation system. This application permits the use of a small attitude error model implementation in the alignment Kalman filter. The second subcase is an alignment without initial attitudes. This application requires a reformulation of the error model to accommodate large attitude errors. Both approaches are illustrated using results obtained with actual recorded flight test data.

The next case study is for a differential GPS- (DGPS) aided dead-reckoning (DR) land navigation system and is presented in Chapter 14. In this case study, higher rate navigation data, as required by an autonomous vehicle for real-time steering and control, are provided at a comparable level of accuracy as available from the lower-rate DGPS position outputs. The DR navigation solution is formed using outputs from a single-axis gyro and a Doppler radar ground speed sensor. In this case study, test data from an actual real-time implementation are used to illustrate the benefits and performance available from this integration.

In Chapter 15, attitude determination and estimation case studies are presented. These case studies address navigation problems, like those of the previous chapter, that use less than a complete suite of accelerometer and gyro sensors. The problem of attitude determination involves determining a transformation matrix that transforms known information from one reference frame to another. This problem has applications in both terrestrial and spacecraft navigation. Attitude estimation continues with the integration of a navigation solution, in this case only attitude, with other sensor data via a Kalman filter. Examples of these two applications are presented.

Summarized in Chapter 16 are several design options available for an integrated navigation system based on one of many possible navigation frame implementations. These options illustrate various navigation system error models discussed and developed in this book that are candidates for implementation in a Kalman filter algorithm. Mathematical methods presented provide the reader, whether involved in design, integration, or test and evaluation, the tools to establish a new integrated navigation system or evaluate an existing one.

Appendices contain additional material to supplement the material presented in the earlier chapters. This additional material is included to expand upon the material presented in the chapters and to help the book be more of a stand-alone document.

While treating specific subjects, notation is used that is consistent with publications relating to those subjects. However, because a variety of subjects are covered, the same notational variables assume different meanings in various parts of the book. The reader is cautioned not to assume that the symbology in one subject area has the same meaning in another. Finally, a practical aspect

of presenting the variety of material limits the depth to which each area can be developed. The reader is encouraged to investigate those areas that are only summarized to the level necessary to understand their application in an integrated navigation system.

**Robert M. Rogers**

June 2007



**Part 1**

**Elements of Integrated Navigation Systems**



# 1 **Introduction**

Navigation systems are used for land, sea, airborne, and space vehicles. These systems provide an operator and/or control system with the necessary information to effect some action in response to data provided by these systems. For example, this action can be a course correction indication for an aircraft pilot or a feedback control signal to guide an autonomous vehicle. These systems incorporate onboard sensors coupled with a computer, permitting self-contained operations with little or no assistance required from sources external to the vehicle.

The core of the navigation system is a set of sensors combined with a computer that can provide a relatively stable and accurate source of navigation data. These systems output navigation state data, which usually include position, velocity, and attitude. As a result of imperfections in navigation sensors and computational errors, errors develop in the navigation state data and grow in time. The host vehicle's operating environment also influences the error growth rate. Long-term error growth is minimized by including other sensors that provide independent and/or redundant navigation data, that is, position, in an integrated system that optimally combines this independent data source with the core navigation system. These independent sensors, referred to as navigation aids, are characterized by long-term error stability, which can complement the short-term error stability of the navigation system's sensors. When combined within a computer algorithm, such as a Kalman filter, errors from both the core sensors and navigation aids can be estimated to reduce the integrated navigation system's errors. The resulting navigation system will exhibit improved performance, even if independent data are used intermittently or are not available for a short time span.

The majority of this book addresses navigation systems that are mechanized with accelerometer and gyro inertial sensors. These inertial sensors provide sensed accelerations (velocity changes over a time interval) and rates (attitude changes over a time interval). Accelerometers and gyros are mounted in orthogonal triad clusters and enclosed within an inertial measurement unit (IMU) to provide three components of acceleration and rate outputs. These outputs are provided to a computer-implemented numerical integration process that computes a navigation solution yielding a complete set of navigation state data, that is, position, velocity, and attitude. These mechanizations are generally referred to as an inertial navigation unit when enclosed within a case that can

be easily removed and replaced. Implementations that include the inertial sensors, computer, and navigation aids are referred to as an inertial navigation system (INS).

Other navigation systems that use fewer than the full three-axis accelerometer and gyro sensors are also presented. These systems include a dead-reckoning system and attitude reference systems. The dead-reckoning system uses speed (distance traveled) and heading sensors to compute a dead-reckoning navigation solution. This solution is less complete and generally provides only position and heading navigation state data. Two attitude reference systems are presented. One uses accelerometers and a magnetic sensor combined with data from a global positioning system (GPS) receiver. The other uses only gyros aided with a strap-down star sensor. Within each of these types, there is a range of integration approaches.

The navigation system's core sensors provide information as a result of movement. These sensors are fixed to the vehicle, that is, a strap-down IMU. Therefore, these sensors provide information about the movement of the vehicle as reference to the vehicle's frame of reference. The information in this reference frame might not be useful to a pilot or guidance system. Therefore, a navigation solution is established in a reference frame that allows its data to be used conveniently and allows data from other sources, that is, navigation aids, to be easily incorporated. This is accomplished by establishing a navigation frame that is relatively fixed. The literature indicates a variety of navigation system reference frames that have been used in integrated navigation systems. Examples include inertial (stellar referenced), Earth-referenced (north-referenced azimuth), and wander azimuth (free azimuth movement). Even within these examples, there are additional levels of definition, for example, which axis is aligned with what direction and what order one axis is rotated with respect to another. The navigation frame selection can be arbitrary, at the discretion of the designer, or the frame's definition can be specified. In either case, the ability to establish mathematical methods to convert data from one axis definition to another is of primary importance in integrated navigation systems. Methods for establishing reference frames, transforming information between frames, and describing motion in these frames are presented using many navigation reference frames for illustration.

To enhance reliability, navigation systems have been implemented using multiple navigation units or implementations that incorporate both inertial and dead-reckoning navigation solutions. For example, some commercial and military aircraft have two or more inertial units, each operating simultaneously. If a unit fails during flight, then the other unit's outputs are available. In some cases, three units are installed, allowing their outputs to be compared. If one unit is much different from the other two, then that unit might be faulty and is no longer considered valid (inoperative). If two navigation units are available, the outputs from two units can be combined with outputs from available navigation aids to form two independent navigation systems. The outputs from these two independent systems can be optimally combined to establish a better source of navigation state data.

The use of several inertial units can be too costly or not feasible for other reasons; however, other sensors that can be used to enhance system reliability

might be available. If, for example, airspeed and heading sensors are available to implement a dead-reckoning navigation solution, then this implementation's outputs can be optimally combined with navigation aids to yield an improved integrated dead-reckoning navigation system. Implementations such as these are used as the only navigation systems in many applications. Dead-reckoning implementations are generally less accurate than inertial systems—not just as a result of the quality of the sensors used but also as a result of the simplifying assumptions used. When included as part of an integrated navigation system that also includes an INS, the dead-reckoning navigation solution represents a fall-back source of navigation data if the primary source, that is, the INS, becomes inoperative.

Whether the navigation system is based on inertial sensors or dead-reckoning sensors, multiple sources of redundant information—navigation aids—can be optimally combined within a single integrated navigation system to improve its navigation state data. Navigation aids that provide independent sources of position are available, that is, the GPS. Other navigation aids include sources of velocity, for example, Doppler and GPS; sources of relative position to ground-based stations, that is, tactical air navigation; and sources of position relative to fixed landmarks, for example, airborne radar and optical line-of-sight sensors. Combining data from multiple navigation aids with the core navigation system can not only enhance the integrated system's reliability but also improve the quality of resulting navigation state data. Methods for integrating different navigation aids to an integrated navigation system are presented.

A fully integrated navigation system structured to provide a high level of reliability is one that implements a failure mode hierarchy to ensure that the most accurate navigation solution is implemented with whatever sensors remain operative. Fully integrated systems such as these can include both inertial-based navigation systems (including more than one inertial unit) and dead-reckoning-based navigation systems. Whether single or multiple inertial units are used, a dead-reckoning implementation is used (by itself or as part of a structured navigation system), or multiple sources of navigation aid data are available, the methods presented in this book are used in integrated navigation systems such as these.

Integrated navigation systems incorporate elements from diverse fields of study, including kinematics, inertial and navigation aid sensors, and optimal estimation. Integration of navigation systems with navigation aids into an integrated navigation system requires the development of mathematical descriptions of reference frames, descriptions of motion in those frames (kinematics), and corresponding models for navigation aids. This book develops a mathematical framework used to accomplish this integration and builds upon the inertial navigation systems analysis approach by Britting [1]. The optimal combination of navigation sensors/systems and navigation aids is accomplished with the Kalman filter algorithm. For the reader without prior academic experience, the approach used in developing this algorithm is based on the weighted recursive least-squares approach of Sage [2].

Kinematic equations used to describe motion are, in general, nonlinear time varying differential equations. Equations describing the relationship between this motion and navigation aid data are also generally nonlinear and time

varying. To facilitate the implementation of an integrated navigation system via the Kalman filter algorithm, navigation state equations are linearized to establish dynamic equations of navigation state errors. The literature indicates that error representations used to express navigation state errors are varied and nonunique. The diversity of choices available precludes the definition of a single system dynamic and measurement model for an integrated navigation system. At one level, the choice is the integrated navigation system's navigation reference frame selection. At another level, given the navigation frame's definition, is the choice of which of several assumptions to use in establishing error equations for implementation within the Kalman filter algorithm. Because these choices are design decisions, methods to establish different error equations based on defined assumptions are presented.

The goal of this book is to provide the reader, whether involved with design, integration, or test and evaluation, with the necessary mathematical tools to evaluate an integrated navigation system's design.

## Mathematical Preliminaries

This chapter presents a review of the mathematical techniques used in this study of integrated navigation systems. The material presented is brief; therefore, the reader is encouraged to seek more in-depth information than that presented in this review.

The following subjects are treated in this chapter: vector/matrix algebra, vector/matrix calculus, linearization techniques, direction cosine matrices, and miscellaneous mathematical topics. The basic algebraic vector and matrix operations, for example, addition, subtraction, multiplication, and inversion (division), are summarized.

Next, vector and matrix calculus, including differentiation of scalars and vectors with respect to scalars and vectors, is reviewed and applied to the development of a least-squares estimate. This is also the approach used later for developing the Kalman filter algorithm. The calculus of vectors and matrices continues with the linearization of nonlinear equations to obtain linearized equations for perturbations about a nominal solution. The conventional Kalman filter algorithm assumes a linear system, and, used frequently in integrated navigation systems, the extended Kalman filter also requires certain linearized terms from this process (see Chapter 13).

The establishment of coordinate systems and transformations of vector components between different systems, using direction cosine matrices, is presented next. The dynamic equation that describes the evolution of this matrix, which is used extensively in this treatment of integrated navigation systems, is also developed. Finally, presented are miscellaneous mathematical topics, including the angular velocity addition theorem and quaternions.

In this chapter, and subsequent chapters that compose Part 1 of this book, problems are included, most with step-by-step instructions that, if followed, lead to the solution found in the problem statement. The purpose of these problems is to expand on the material presented, and they are drawn from actual experiences in integrated navigation system applications.

### 2.1 Vector/Matrix Algebra

The use of vectors and matrices greatly facilitates the development and compact expression of equations and algorithms used in integrated navigation systems. Their use permits results to be expressed in a general form that has

wide applicability without resorting to expressing results in vector component form. Our use requires only a limited amount of vector/matrix algebra.

### 2.1.1 State-Space Vector

A vector is an arrangement of scalar quantities with governing manipulation rules. A column vector  $\mathbf{x}$  of “ $n$ ” scalar quantities  $x_i$  is expressed as

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (2.1)$$

The transpose of the column vector in Eq. (2.1) is expressed as

$$\mathbf{x}^T = [x_1 \quad x_2 \quad \cdots \quad x_n] \quad (2.2)$$

This is in the form of a row vector.

### 2.1.2 Matrices

A vector can also be considered as an  $n \times 1$  matrix. The matrix  $A$  of  $n \times m$  scalar quantities  $a_{ij}$  is expressed as

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \quad (2.3)$$

This matrix can also be considered as an arrangement of vectors. The transpose of a matrix is obtained by interchanging its rows and columns and is expressed as

$$A^T = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ a_{1m} & a_{2m} & \cdots & a_{mm} \end{bmatrix} \quad (2.4)$$

### 2.1.3 Vector/Matrix Manipulations

Compatible vectors and matrices are manipulated according to the following elementary operations:

Addition:

$$C = A + B \quad \text{or} \quad c_{ij} = a_{ij} + b_{ij} \quad (2.5)$$

Subtraction:

$$C = A - B \quad \text{or} \quad c_{ij} = a_{ij} - b_{ij} \quad (2.6)$$

Multiplication:

$$C = AB \quad \text{or} \quad c_{ij} = \sum_{k=1}^n a_{ik} * b_{kj} \quad (2.7)$$

In addition and subtraction, compatible vectors/matrices have the same number of rows and columns. For multiplication, the number of columns of the left matrix must equal the number of rows of the right matrix.

Matrix *inversion* is the parallel for the division operation. The inverse of the square  $n \times n$  matrix  $A$  is defined such that

$$AA^{-1} = A^{-1}A = I_{n \times n} \quad (2.8)$$

where the identity matrix  $I_{n \times n}$  is defined as

$$I_{n \times n} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdot & \cdot & \ddots & \cdot \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

The matrix inverse is defined if the matrix to be inverted is nonsingular. A test to determine if a matrix is nonsingular is that its determinate is nonzero:

$$|A| \neq 0 \quad (2.10)$$

There are special rules for vector/matrix manipulation. First, the order of matrix multiplication is important. In general, matrices do not commute:

$$AB \neq BA \quad (2.11)$$

The transpose of a product is the product of the transposed matrices in reverse order:

$$(AB) = B^T A^T \quad (2.12)$$

Examples of the preceding operations are presented in the following. The inner product (three-space vector dot product parallel) of two vectors is

$$\mathbf{x}^T \mathbf{y} = \sum_{i=1}^n x_i y_i = x_1 y_1 + x_2 y_2 + \cdots + x_n y_n \quad (2.13)$$

The outer product of a  $n \times 1$  vector  $\mathbf{x}$  and a  $m \times 1$  vector  $\mathbf{y}$  is

$$\mathbf{xy}^T = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \begin{bmatrix} y_1 & y_2 & \cdots & y_m \end{bmatrix} = \begin{bmatrix} x_1y_1 & x_1y_2 & \cdots & x_1y_m \\ x_2y_1 & x_2y_2 & \cdots & x_2y_m \\ \vdots & \vdots & \ddots & \vdots \\ x_ny_1 & x_ny_2 & \cdots & x_ny_m \end{bmatrix} \quad (2.14)$$

*Matrix/vector product.* The product of a compatible vector and matrix is given as

$$\mathbf{y} = A\mathbf{x} \quad (2.15)$$

or, written out in component form and with multiplication completed,

$$\begin{aligned} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} &= \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \\ &= \begin{bmatrix} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \end{bmatrix} \end{aligned} \quad (2.16)$$

Reviewing the condition of compatible operations from this example, an  $m \times 1$  vector  $\mathbf{y}$  results from the product of a  $m \times n$  matrix  $A$  and an  $n \times 1$  vector  $\mathbf{x}$ . The number of columns of the matrix must equal the number of rows of the vector multiplied. Products between matrices must also be compatible.

*Similarity transformation.* In Eq. (2.15),  $\mathbf{x}$  and  $\mathbf{y}$  are related as  $\mathbf{y} = A\mathbf{x}$ . If the vectors  $\mathbf{x}$  and  $\mathbf{y}$  are transformed into another set of vectors by a nonsingular invertible matrix  $B$  as

$$\mathbf{u} = A\mathbf{x} \quad (2.17)$$

$$\mathbf{v} = B\mathbf{y} \quad (2.18)$$

then the vectors  $\mathbf{u}$  and  $\mathbf{v}$  are related by a similarity transformation. This is demonstrated in the following steps:

$$\mathbf{v} = B\mathbf{y} = B(A\mathbf{x}) = BAB^{-1}\mathbf{u} \quad (2.19)$$

where the matrix  $(BAB^{-1})$  is the similarity transformation.

### 2.1.4 Three-Space Vectors

Certain operations specialized to three-dimensional “space” vectors will be used. The vector cross product can be expressed as a matrix/vector product.

The cross product of two vectors  $\mathbf{a}$  and  $\mathbf{b}$  is expressed as

$$\begin{aligned}
 \mathbf{a} \times \mathbf{b} &= \begin{bmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{bmatrix} \\
 &= \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \equiv (\mathbf{a} \times) \mathbf{b} \\
 &= \begin{bmatrix} 0 & b_3 & -b_2 \\ -b_3 & 0 & b_1 \\ b_2 & -b_1 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \equiv -(\mathbf{b} \times) \mathbf{a} \\
 &= -\mathbf{b} \times \mathbf{a}
 \end{aligned} \tag{2.20}$$

These forms demonstrate that a vector cross product can be formed as the product of a matrix and a vector where the matrix is *skew-symmetric*. A skew-symmetric matrix has the property that its transpose is the negative of the matrix, that is,

$$A^T = -A \tag{2.21}$$

The following vector *triple cross* product is applied in later developments:

$$\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) + \mathbf{b} \times (\mathbf{c} \times \mathbf{a}) + \mathbf{c} \times (\mathbf{a} \times \mathbf{b}) = 0 \tag{2.22}$$

### Example 2.1 U–D Factorization of Symmetric $3 \times 3$ P Matrix

Assume that a symmetric matrix (a matrix whose elements in opposite pairings about the matrix's diagonal are equal)  $P$  can be factored into the following form:

$$P = UDU^T$$

where the  $U$  matrix is an upper triangular matrix with unity diagonal elements, the  $D$  matrix is a diagonal matrix with nonzero diagonal entries, and all other elements are zero. Then, what are the relationships between the original symmetric  $P$  matrix elements and the elements of the  $U$  and  $D$  matrices so defined?

For the simple  $3 \times 3$  matrix, this equation is written in expanded form as

$$\begin{aligned}
 \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{12} & p_{22} & p_{23} \\ p_{13} & p_{23} & p_{33} \end{bmatrix} &= \begin{bmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} d_1 & 0 & 0 \\ 0 & d_2 & 0 \\ 0 & 0 & d_3 \end{bmatrix} U^T \\
 &= \begin{bmatrix} d_1 + u_{12}d_2u_{12} + u_{13}d_3u_{13} & u_{12}d_2 + u_{13}d_3u_{23} & u_{13}d_3 \\ d_2u_{12} + u_{23}d_3u_{13} & d_2 + u_{23}d_3u_{23} & u_{23}d_3 \\ d_3u_{13} & d_3u_{23} & d_3 \end{bmatrix}
 \end{aligned}$$

Equating elements in reverse order, the following are obtained:

$$d_3 = p_{33}$$

$$u_{23} = \frac{p_{23}}{d_3}$$

$$u_{13} = \frac{p_{13}}{d_3}$$

$$d_2 = p_{22} - u_{23}d_3u_{23}$$

$$u_{12} = \frac{p_{12} - u_{13}d_3u_{23}}{d_2}$$

and

$$d_1 = p_{11} - (u_{12}d_2u_{12} + u_{13}d_3u_{13})$$

The preceding equations can be summarized in the following algorithm [3]:

```
for  $j = n, n - 1, \dots, 2$ 
   $d_j = p_{jj}$ 
  for  $k = 1, \dots, j - 1$ 
     $u_{kj} = p_{kj}/d_j$ 
    for  $i = 1, \dots, k$ 
       $p_{ik} = p_{ik} - u_{ij} d_j u_{kj}$ 
and
 $d_1 = p_{11}$ 
```

In this particular form for the  $U-D$  factorization algorithm form, the upper diagonal elements of the  $P$  matrix are destroyed.

### Example 2.2 Matrix Inversion Lemma

This lemma [2] is used in later developments for the Kalman filtering algorithm and is presented here to illustrate the algebra of matrix manipulations.

This lemma is stated as follows: if  $A$ ,  $B$ , and  $D$  are invertible matrices such that

$$A^{-1} = B^{-1} + C^T D^{-1} C$$

and

$$(D + CBC^T)^{-1}$$

exists, then

$$A = B - BC^T(D + CBC^T)^{-1}CB$$

The following demonstration of the proof of this is summarized from [2].

Step 1: Premultiply the first equation by  $A$ :

$$A[A^{-1} = B^{-1} + C^T D^{-1} C] \implies I = AB^{-1} + AC^T D^{-1} C$$

Step 2: Postmultiply this result by  $B$ :

$$[I = AB^{-1} + AC^T D^{-1} C]B \implies B = A + AC^T D^{-1} CB$$

Step 3: Postmultiply this result by  $C^T$ :

$$\begin{aligned} [B = A + AC^T D^{-1} CB]C^T &\implies BC^T = AC^T + AC^T D^{-1} CBC^T \\ &= AC^T D^{-1} D + AC^T D^{-1} CBC^T \\ &= AC^T D^{-1} (D + CBC^T) \end{aligned}$$

Step 4: Postmultiply by  $(D + CBC^T)^{-1}$ :

$$[BC^T = AC^T D^{-1} (D + CBC^T)]^{-1} \implies BC^T (D + CBC^T)^{-1} = AC^T D^{-1}$$

Step 5: Postmultiply by  $CB$ :

$$\begin{aligned} [BC^T (D + CBC^T)^{-1} = AC^T D^{-1}]CB &\implies BC^T (D + CBC^T)^{-1} CB \\ &= AC^T D^{-1} CB \end{aligned}$$

Step 6: Subtracting both sides from  $B$  yields

$$B - BC^T (D + CBC^T)^{-1} CB = B - AC^T D^{-1} CB$$

Step 7: Using the result of step 2 for  $B$  on the right-hand side (RHS),

$$B - BC^T (D + CBC^T)^{-1} CB = A + AC^T D^{-1} CB - AC^T D^{-1} CB = A$$

which is the desired result.

## 2.2 Vector/Matrix Calculus

The calculus of vectors and matrices follows similar manipulation rules. A few of these manipulations are reviewed in this section.

The derivative of a vector (matrix) with respect to a scalar is a vector (matrix) whose elements are derivatives with respect to that scalar. (In this example, the scalar is time- $t$ .)

$$\frac{d\mathbf{x}}{dt} = \begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \\ \vdots \\ \frac{dx_n}{dt} \end{bmatrix} \quad (2.23)$$

The derivative of a scalar with respect to a vector is defined as the row vector

$$\frac{\partial a}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial a}{\partial x_1} & \frac{\partial a}{\partial x_2} & \cdots & \frac{\partial a}{\partial x_n} \end{bmatrix} \quad (2.24)$$

If the scalar  $a$  is defined as

$$a = \mathbf{y}^T \mathbf{x} = y_1 x_1 + y_2 x_2 + \cdots + y_n x_n \quad (2.25)$$

then

$$\frac{\partial a}{\partial \mathbf{x}} = [y_1 \quad y_2 \quad \cdots \quad y_n] \quad (2.26)$$

The derivative of a vector with respect to another vector is defined as the matrix. For example, if the vector  $\mathbf{y}$  is defined as  $\mathbf{y} = A\mathbf{x}$ , or written out in component form as in Eq. (2.16),

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \end{bmatrix} \quad (2.27)$$

From the definition of the derivative of a scalar with respect to a vector, the first component becomes

$$\frac{\partial y_1}{\partial \mathbf{x}} = [a_{11} \quad a_{12} \quad \cdots \quad a_{1n}] \quad (2.28)$$

The second is

$$\frac{\partial y_2}{\partial \mathbf{x}} = [a_{21} \quad a_{22} \quad \cdots \quad a_{2n}] \quad (2.29)$$

Continuing for each element in the vector  $\mathbf{y}$ , then ordering the results into a vector, the result is summarized in the following vector/matrix form

$$\frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}} (A\mathbf{x}) = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = A \quad (2.30)$$

### Example 2.3 Least-Squares Parameter Estimation

The objective of this example is to develop an expression for an optimal estimate of a vector composed of unknown constants given available data that are related to those constants, that is, polynomial curve fitting.

Given:

$\mathbf{x}$ -vector of  $n$  unknown constant parameters

$\mathbf{z}$ -vector of  $m$  available data quantities from which to estimate  $\mathbf{x}$

Only indirect and imperfect access to the unknown constants is available via the measurements  $\mathbf{z}$ . The vector  $\mathbf{z}$  is linearly related to  $\mathbf{x}$ , but each element  $z$  is corrupted with additive error  $\mathbf{v}$  as

$$\mathbf{z} = H\mathbf{x} + \mathbf{v} \quad (2.31)$$

Here,  $H$  is an  $m \times n$  known matrix, where  $m \geq n$  (more measurements than unknown parameters) and with rank  $n$  ( $n$  linearly independent rows or columns) and  $\mathbf{v}$  is a vector of  $m$  unknown additive errors.

Form a cost or penalty function that is an inner product as shown here:

$$J_{LS} = (\mathbf{z} - H\mathbf{x})^T(\mathbf{z} - H\mathbf{x}) \quad (2.32)$$

Then, the objective is to minimize the difference, in a least-squares sense, between the available data in the vector  $\mathbf{z}$  and an estimate of these data based on estimates for unknown constant contained within the vector  $\mathbf{x}$  and the known matrix  $H$ .

The estimate for the vector  $\mathbf{x}$  is obtained by minimizing  $J_{LS}$  with respect to  $\mathbf{x}$ . An extrema (maximum or minimum) is obtained by setting the derivative of  $J_{LS}$  with respect to  $\mathbf{x}$  to zero and solving for  $\mathbf{x}$ . First expanding Eq. (2.32),

$$J_{LS} = \mathbf{z}^T \mathbf{z} - \mathbf{z}^T H\mathbf{x} - \mathbf{x}^T H^T \mathbf{z} + \mathbf{x}^T H^T H\mathbf{x}$$

and then, the derivative of this scalar with respect to  $\mathbf{x}$  is obtained in the following sequence:

$$\begin{aligned} \frac{\partial J_{LS}}{\partial \mathbf{x}} &= 0^T - \mathbf{z}^T H - (H^T \mathbf{z})^T + (H^T H\mathbf{x})^T + \mathbf{x}^T H^T H \\ &= -2(\mathbf{z}^T H + \mathbf{x}^T H^T H) \end{aligned}$$

Setting this result to zero for an extrema, canceling the 2, and transposing the entire equation yields

$$H^T H\mathbf{x} = H^T \mathbf{z}$$

Because it is assumed that the matrix  $H$  is of full rank, the matrix resulting from the product  $(H^T H)$  is invertible.

The expression for  $\mathbf{x}$  that minimizes or maximizes the cost function becomes

$$\hat{\mathbf{x}} = (H^T H)^{-1} H^T \mathbf{z} \quad (2.33)$$

This expression produces a minimum value of  $J_{LS}$  because the second derivative

$$\frac{\partial^2 J_{LS}}{\partial \mathbf{x}^2} = 2(H^T H) \quad (2.34)$$

is a positive (definite) quantity.

Important properties associated with the estimate just obtained are listed here:

1) Estimation error:

$$\begin{aligned}
 \boldsymbol{\varepsilon} &\equiv \mathbf{x} - \hat{\mathbf{x}} \\
 &= \mathbf{x} - (H^T H)^{-1} H^T \mathbf{z} \\
 &= \mathbf{x} - (H^T H)^{-1} H^T (H\mathbf{x} + \boldsymbol{\nu}) \\
 &= \mathbf{x} - (H^T H)^{-1} (H^T H)\mathbf{x} - (H^T H)^{-1} H^T \boldsymbol{\nu} \\
 &= -(H^T H)^{-1} H^T \boldsymbol{\nu}
 \end{aligned} \tag{2.35}$$

2) Residual (difference between actual and predicted data):

$$\begin{aligned}
 \boldsymbol{\nu} &\equiv H\mathbf{x} + \boldsymbol{\nu} - H\hat{\mathbf{x}} \\
 &= H(\mathbf{x} - \hat{\mathbf{x}}) + \boldsymbol{\nu} \\
 &= H\boldsymbol{\varepsilon} + \boldsymbol{\nu} \\
 &= -H(H^T H)^{-1} H^T \boldsymbol{\nu} + \boldsymbol{\nu} \\
 &= [I - H(H^T H)^{-1} H^T] \boldsymbol{\nu}
 \end{aligned} \tag{2.36}$$

If errors  $\boldsymbol{\nu}$  are random zero mean processes such that

$$E[\boldsymbol{\nu}] \equiv 0 \quad \text{and} \quad E[\boldsymbol{\nu} \boldsymbol{\nu}^T] \equiv R \tag{2.37}$$

then the following additional properties are obtained.

3) Expected estimation error:

$$\begin{aligned}
 E[\mathbf{x} - \hat{\mathbf{x}}] &= -E[(H^T H)^{-1} H^T \boldsymbol{\nu}] \\
 &= -(H^T H)^{-1} H^T E[\boldsymbol{\nu}] \\
 &= 0
 \end{aligned} \tag{2.38}$$

The estimation error is zero mean. The mathematical expectation operator  $E[ ]$  is used extensively in this book. Its development is beyond the scope of this book. However, because all errors considered in this book are Gaussian, this operator can be considered to be operating on a random variable yielding a known mean and variance.

4) Estimation error covariance:

$$\begin{aligned}
 E[(\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T] &= E\{[(H^T H)^{-1} H^T \boldsymbol{\nu}][(H^T H)^{-1} H^T \boldsymbol{\nu}]^T\} \\
 &= (H^T H)^{-1} H^T E[\boldsymbol{\nu} \boldsymbol{\nu}^T] H (H^T H)^{-T} \\
 &= (H^T H)^{-1} H^T R H (H^T H)^{-T} \\
 &\equiv P
 \end{aligned} \tag{2.39}$$

5) Residual error covariance:

$$\begin{aligned}
 E[(z - H\hat{x})(z - H\hat{x})^T] &= [I - H(H^T H)^{-1}H^T]E[v v^T] \\
 &\quad \times [I - H(H^T H)^{-1}H^T]^T \\
 &= [I - H(H^T H)^{-1}H^T]R \\
 &\quad \times [I - H(H^T H)^{-1}H^T]^T
 \end{aligned} \tag{2.40}$$

### 2.3 Linearization Techniques

The equations describing terrestrial navigation are nonlinear differential equations. Linearized error forms are desirable for use in the Kalman filter algorithm. In this section, techniques are presented to obtain a linearized error form of general nonlinear equations.

Consider the following continuous-time nonlinear vector differential equation:

$$\dot{x} = f(x, t) \tag{2.41}$$

where the dot over the variable denotes the derivative with respect to the scalar  $t$ , time. Assume that a nominal solution  $x_0$  satisfies this equation. Examine the perturbations or errors  $\delta x$  about this nominal solution with time fixed, where the new value of  $x$  is defined as

$$x = x_0 + \delta x \tag{2.42}$$

Taking the time derivative of this equation yields

$$\dot{x} = \dot{x}_0 + \delta\dot{x}$$

Upon substitution, the original differential equation becomes

$$\dot{x}_0 + \delta\dot{x} = f(x_0 + \delta x, t)$$

Expand  $f(x_0 + \delta x, t)$  in a Taylor-series about the nominal solution  $x_0$  and for time  $t$  fixed:

$$f(x_0 + \delta x, t) = f(x_0, t) + \frac{\partial f(x, t)}{\partial x} \Big|_{x=x_0} \delta x + \text{HOT} \tag{2.43}$$

where HOT represents higher-order terms beyond the first linear term that are not included. The original differential equation is then expressed as

$$\dot{x}_0 + \delta\dot{x} \approx f(x_0, t) + \frac{\partial f(x, t)}{\partial x} \Big|_{x=x_0} \delta x$$

Because  $\dot{\mathbf{x}}_0$  is also a solution to the original differential equation,

$$\dot{\mathbf{x}}_0 = \mathbf{f}(\mathbf{x}_0, t) \quad (2.44)$$

By subtracting, the linearized equation for the continuous-time error becomes

$$\delta\dot{\mathbf{x}} = \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{x}_0} \delta\mathbf{x} \quad (2.45)$$

In this equation, only first-order terms have been retained from the Taylor-series expansion.

A concept used in the development of the dynamic Kalman filter equations is the state transition matrix. The preceding differential equation is used to establish the values of the linearized state  $\delta\mathbf{x}$  at a future time instant  $t + \Delta t$ . With the assumption of a sufficiently small time step  $\Delta t$ , such that a Taylor-series time expansion retains only the first-order term, the continuous-time error equation can be rewritten as

$$\begin{aligned} \delta\mathbf{x}(t + \Delta t) &\approx \delta\mathbf{x}(t) + \delta\dot{\mathbf{x}}(t)\Delta t \\ &= \delta\mathbf{x}(t) + \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \delta\mathbf{x}(t)\Delta t \end{aligned}$$

or

$$\delta\mathbf{x}(t + \Delta t) \approx \left( I + \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Delta t \right) \delta\mathbf{x}(t) \quad (2.46)$$

Defining the matrix in parentheses as the state transition matrix, this equation can be rewritten as

$$\delta\mathbf{x}(t + \Delta t) \equiv \Phi_{t+\Delta t} \delta\mathbf{x}(t) \quad (2.47)$$

where

$$\Phi_{t+\Delta t} \approx I + \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Delta t \quad (2.48)$$

More formal definitions and methods of determining this matrix exist. The partial derivative in this equation will later be referred to by the letter  $F$ :

$$F \equiv \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \quad (2.49)$$

With this notation, the state transition matrix becomes

$$\Phi_{t+\Delta t} \approx I + F \Delta t \quad (2.50)$$

A more formal derivation of the state transition matrix, for linear time-invariant systems, shows that this matrix is the exponential matrix  $e^{F\Delta t}$ , and the form in Eq. (2.50) is just a truncation after the first term in the exponential expansion

$$\begin{aligned}\Phi_{t+\Delta t} &= e^{F\Delta t} \\ &= I + F\Delta t + F^2 \frac{\Delta t^2}{2!} + F^3 \frac{\Delta t^3}{3!} + F^4 \frac{\Delta t^4}{4!} + \dots\end{aligned}\quad (2.51)$$

Improved accuracy can be obtained by retaining more terms, however, with a higher computational cost.

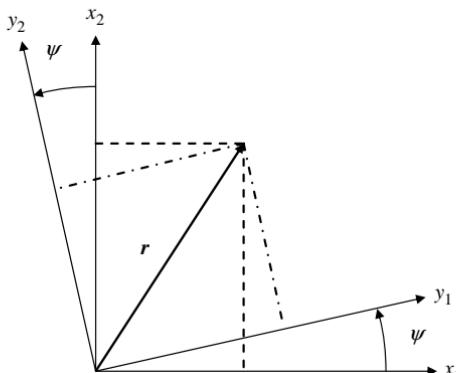
## 2.4 Direction Cosine Matrices

### 2.4.1 Coordinate Transformations

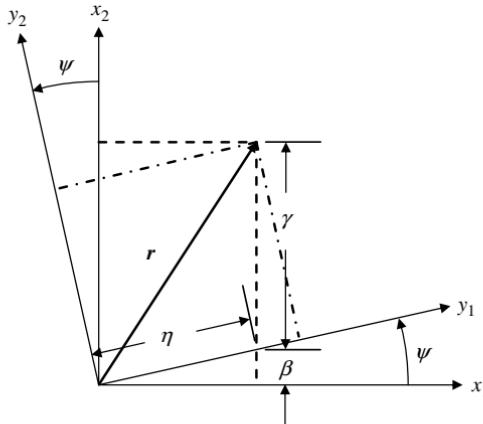
Important matrices used in this treatment of integrated navigation systems are direction cosine matrices (DCMs). These matrices relate a vector's components in one coordinate frame to another frame. The following development illustrates how this matrix's elements are formed by considering the transformation of a vector's components, from an original coordinate frame, into components defined in another frame, with the second frame being rotated relative to the original.

Consider the vector  $\mathbf{r}$  shown in Fig. 2.1. In the coordinate frame  $x$ , this vector is denoted as  $\mathbf{r}^x$ . In the rotated frame  $y$ , this vector is unchanged in length and is denoted as  $\mathbf{r}^y$ . It is desired to determine the component values associated with this vector in the rotated frame  $y$ . This frame is rotated with respect to the  $x$  frame by the angle  $\psi$ .

Figure 2.1 is redrawn in Fig. 2.2 with emphasis on a few triangle segments used to develop expressions for the vector components. Using simple



**Fig. 2.1** Vector/coordinate frame.



**Fig. 2.2 Vector/coordinate frame components.**

trigonometry, the following relationships are established:

$$x_1 = \eta \cos \psi \implies \eta = \frac{x_1}{\cos \psi}$$

$$\beta = \eta \sin \psi \implies \beta = \frac{x_1 \sin \psi}{\cos \psi}$$

$$y_1 - \eta = \gamma \sin \psi = (x_2 - \beta) \sin \psi$$

$$y_2 = \gamma \cos \psi = (x_2 - \beta) \cos \psi$$

$$y_1 = x_2 \sin \psi - \beta \sin \psi + \eta$$

$$y_2 = x_2 \cos \psi - \beta \cos \psi$$

$$y_1 = x_2 \sin \psi - x_1 \frac{\sin \psi}{\cos \psi} \sin \psi + \frac{x_1}{\cos \psi}$$

$$y_2 = x_2 \cos \psi - x_1 \frac{\sin \psi}{\cos \psi} \cos \psi$$

$$y_1 = x_1 \left[ \frac{(1 - \sin^2 \psi)}{\cos \psi} \right] + x_2 \sin \psi$$

$$y_2 = -x_1 \sin \psi + x_2 \cos \psi$$

or

$$y_1 = \cos \psi x_1 + \sin \psi x_2 \quad (2.52)$$

$$y_2 = -\sin \psi x_1 + \cos \psi x_2 \quad (2.53)$$

The preceding two equations are written in vector/matrix form as

$$\mathbf{y} = \mathbf{C}\mathbf{x} \quad (2.54)$$

where the transformation matrix  $\mathbf{C}$  is

$$\mathbf{C} = \begin{bmatrix} \cos \psi & \sin \psi \\ -\sin \psi & \cos \psi \end{bmatrix} \quad (2.55)$$

This matrix relates vector components in the  $x$  frame to components in the  $y$  frame.

Direction cosine matrices have several important properties. First, the determinant is unity,

$$|\mathbf{C}| = 1 \quad (2.56)$$

Next, the transpose is equal to the inverse,

$$\mathbf{C}^T = \mathbf{C}^{-1} \quad (2.57)$$

Finally, the rows and columns of a DCM form orthogonal unit vectors. The dot product of two rows or columns of this matrix is zero, and the cross product of two rows or columns of the matrix yields the remaining row or column. These properties are illustrated next with the simple DCM just given.

Determinant:

$$\begin{aligned} |\mathbf{C}| &= \begin{vmatrix} \cos \psi & \sin \psi \\ -\sin \psi & \cos \psi \end{vmatrix} \\ &= \cos^2 \psi + \sin^2 \psi = 1 \end{aligned}$$

Inverse:

$$\begin{aligned} \mathbf{C}^T \mathbf{C} &= \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix} \begin{bmatrix} \cos \psi & \sin \psi \\ -\sin \psi & \cos \psi \end{bmatrix} \\ &= \begin{bmatrix} \cos^2 \psi + \sin^2 \psi & -\cos \psi \sin \psi + \sin \psi \cos \psi \\ -\sin \psi \cos \psi + \cos \psi \sin \psi & \sin^2 \psi + \cos^2 \psi \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned}$$

Orthogonality:

$$\begin{bmatrix} \cos \psi & \sin \psi \end{bmatrix} \begin{bmatrix} -\sin \psi \\ \cos \psi \end{bmatrix} = -\cos \psi \sin \psi + \sin \psi \cos \psi = 0$$

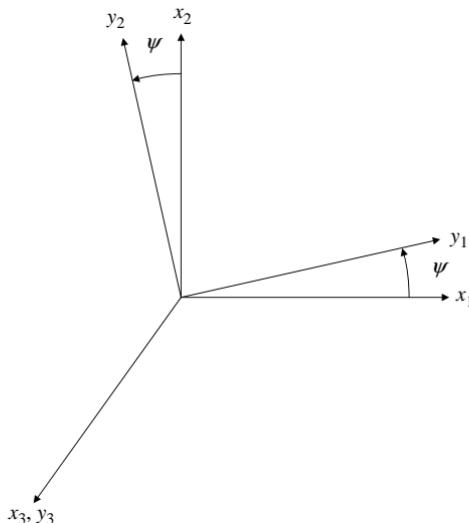
Coordinate frames used in this treatment of navigation systems are defined in a *right-handed* sense. That is, the direction of rotation formed by curling ones fingers is in the direction of the thumb on the right hand. In the examples in Figs. 2.1 and 2.2, the rotation was about a third axis perpendicular to the plane of the paper. This third axis completes the right-handed Cartesian frame. The transformed vector component along this third axis remains unchanged with the rotation about that axis. This is illustrated in Fig. 2.3.

For the case of three axes, the earlier  $C$  matrix is rewritten as

$$C = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.58)$$

Several features about this matrix and the way it is established can be identified from its form just given. First, elements of the DCM row/column about which the rotation occurs are either 0 or 1. Second, the other elements in the DCM are either sin or cos of the angle of rotation, with cosines being on the diagonal and sines being off the diagonal. Finally, the negative sign on the sine term is associated with the rotated frame's component, which is rotated "outside" of the quadrant formed by the original frame's axes.

A vector's components described in one frame can be described in another frame of arbitrary orientation with respect to the original frame by a transformation matrix composed of three sequential rotations (Euler angles) starting from the original frame's axes. These rotations are illustrated in Fig. 2.4, and the



**Fig. 2.3 Single rotation in three-axis coordinate frame.**

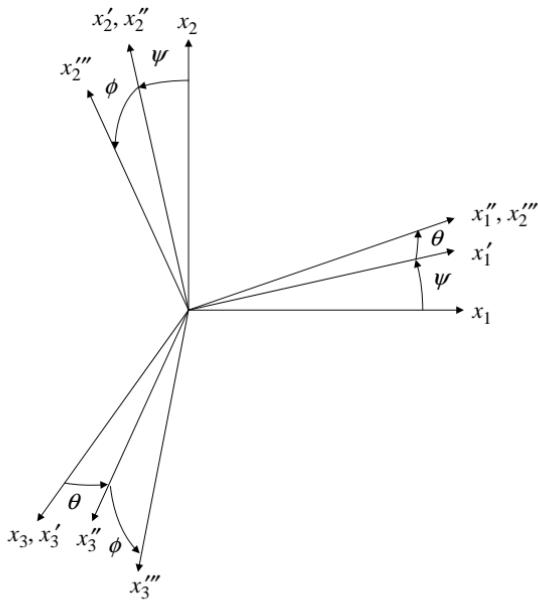


Fig. 2.4 Three rotations in three-axis coordinate frame.

primes are used to represent intermediate axes in the sequence of rotations. The final  $y$  frame corresponds to the triple primed  $x$  axes. Written in vector form, the transformation is

$$\mathbf{r}^y = C_x^y \mathbf{r}^x \quad (2.59)$$

where the transformation DCM  $C_x^y$  transforms the components of the  $\mathbf{r}$  vector from the  $x$  frame to the  $y$  frame.

Mathematically, this transformation is accomplished by transforming through three sequential rotations as shown in Eqs. (2.60–2.62). These rotations proceed as a  $\psi$  rotation about  $x_3$ ; a  $\theta$  rotation about  $x'_2$  resulting from the first rotation; and, finally, a  $\phi$  rotation about  $x''_1$  resulting from the second rotation, where the primes denote the intermediate frames generated by sequential rotations.

Rotation 1:

$$\begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix} = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (2.60)$$

The diagram illustrates the first rotation. It shows the original axes \$x\_1, x\_2, x\_3\$ and the new axes \$x'\_1, x'\_2, x'\_3\$. The angle \$\psi\$ is indicated between the \$x\_1\$ axis and the \$x'\_1\$ axis.

Rotation 2:

$$\begin{bmatrix} x_1'' \\ x_2'' \\ x_3'' \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} x_1' \\ x_2' \\ x_3' \end{bmatrix} \quad (2.61)$$

Rotation 3:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} x_1'' \\ x_2'' \\ x_3'' \end{bmatrix} \quad (2.62)$$

Combining the sequential rotations into a single matrix,

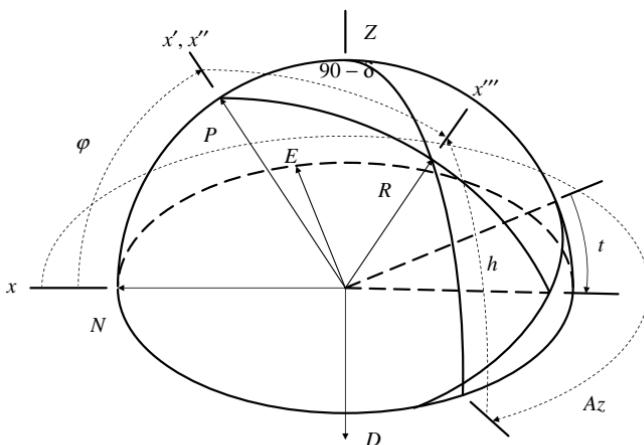
$$\mathbf{r}^y = C_x^y \mathbf{r}^x = C_{x''}^y C_{x'}^{x''} C_x^{x'} \mathbf{r}^x \quad (2.63)$$

The resulting DCM (transposed) is written as

$$C_y^x = \begin{bmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (2.64)$$

#### Example 2.4 Using DCMs to Obtain Local Azimuth and Elevation to Stars

It is desired to relate the astronomical position of a star or some other body to a set of local-level azimuth and elevation angles. With knowledge of the local position (latitude) and the astronomical body's position (local hour angle and declination), the local line-of-sight direction (azimuth and elevation) can be determined. A local north–east–down (*N-E-D*) reference frame is shown in Fig. 2.5. From the origin of this frame, the point *P* is located to correspond to the approximate direction of the star Polaris. This corresponds to the origin's latitude. The point *Z* is in the direction of the zenith directly above the origin. From the meridian formed by the points *N*, *P*, and *Z*, the local hour angle *t* and the declination *δ* establish the astronomical body *R* position. The azimuth *Az* and elevation *h* angles are to be obtained in terms of the local hour angle and declination.



**Fig. 2.5** North–east–down reference frame.

**Astronomical triangle.** One approach is to use the astronomical triangle and spherical triangle trigonometry to establish these relationships. This approach will be illustrated first. The astronomical triangle is formed by the points  $P$ ,  $Z$ , and  $R$  in Fig. 2.5 and is redrawn in Fig. 2.6.

For a spherical triangle with sides  $a$ ,  $b$ , and  $c$ , with opposing angles  $A$ ,  $B$ , and  $C$ ,

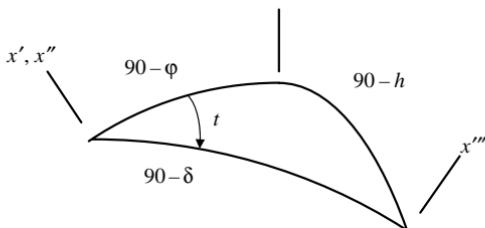
$$\frac{\sin A}{\sin a} = \frac{\sin B}{\sin b} = \frac{\sin C}{\sin c}$$

and

$$\cos a = \cos b \cos c + \sin b \sin c \cos A$$

Identifying the local hour angle  $t$  with  $A$  in this equation, the sides  $a$ ,  $b$ , and  $c$  correspond to  $(90 - h)$ ,  $(90 - \delta)$ , and  $(90 - \phi)$ , respectively, and yield the following equation for elevation:

$$\sin h = \sin \delta \sin \phi + \cos \delta \cos \phi \cos t$$



**Fig. 2.6** Astronomical triangle.

The ratio of sine's above yields

$$\frac{\sin(360 - Az)}{\sin(90 - \delta)} = \frac{\sin t}{\sin(90 - h)}$$

from which the following equation for azimuth is obtained:

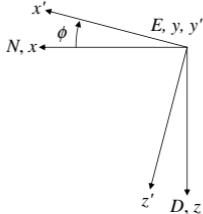
$$\sin Az = \frac{-\cos \delta \sin t}{\cos h}$$

By moving the angle identifications around the triangle, the following alternate form for the azimuth equation can be obtained:

$$\cos Az = \frac{\sin \delta \cos \phi - \cos \delta \sin \phi \cos t}{\cos h}$$

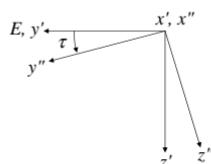
**Direction cosines.** With reference to Fig. 2.5, an original set of axes  $x-y-z$  is aligned parallel to the  $N-E-D$  axes shown. A sequence of three rotations aligns the  $x$  axis with the final orientation, which is along the line of sight to the astronomical object passing through the point  $R$ . First, a rotation about the  $y$  axis, through latitude  $\phi$ , aligns the original  $x$  axis direction with the point  $P$  to form the intermediate  $x'$  axis as

$$C_1 = \begin{bmatrix} c\phi & 0 & -s\phi \\ 0 & 1 & 0 \\ s\phi & 0 & c\phi \end{bmatrix}$$



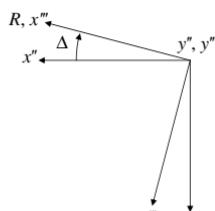
Second, a rotation about the intermediate  $x'$  axis through the hour angle  $t$  is shown

$$C_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & ct & st \\ 0 & -st & ct \end{bmatrix}$$



This is shown as a  $\tau$  rotation, where  $\tau = -t$ . Finally, a rotation about what is now the  $y''$  axis, through  $90 - \delta$ , to align the final  $x'''$  axis with the astronomical body is shown as

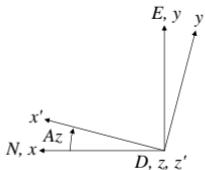
$$C_3 = \begin{bmatrix} c(90 - \delta) & 0 & -s(90 - \delta) \\ 0 & 1 & 0 \\ s(90 - \delta) & 0 & c(90 - \delta) \end{bmatrix}$$



This rotation is shown as a  $\Delta$  rotation, where  $\Delta = 90 - \delta$ .

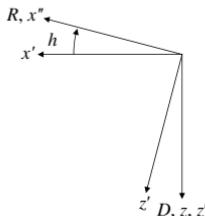
Consider another sequence of two rotations, again starting with the original  $x-y-z$  axes parallel to the  $N-E-D$  axes, however, through different angles to arrive at the same orientation just obtained. First, an azimuth rotation, through  $Az$ , about the  $z$  axis is shown as

$$C_4 = \begin{bmatrix} cAz & sAz & 0 \\ -sAz & cAz & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



This rotation is followed by a rotation, through the elevation angle  $h$ , about the intermediate  $y'$  axis to align the final  $x''$  axes with the astronomical body:

$$C_5 = \begin{bmatrix} ch & 0 & -sh \\ 0 & 1 & 0 \\ sh & 0 & ch \end{bmatrix}$$



Because the sequences just described start with the same  $x-y-z$  axes and end with the alignment of their corresponding final  $x$  axis with astronomical body, the first row's elements of the following matrix product are equivalent:

$$C_3 C_2 C_1 = C_5 C_4$$

Equating 1:3 elements of the resulting matrices yields the following equation for the elevation angle:

$$\sin h = \sin \delta \sin \phi + \cos \delta \cos t \cos \phi$$

which is identical to that obtained by using the astronomical triangle approach. Equating the 1:2 elements of this matrix product yields the following equation for the azimuth angle:

$$\sin Az = \frac{-\cos \delta \sin t}{\cos h}$$

The same expression for the azimuth angle, as just obtained using the astronomical triangle approach, is found by equating the 1:1 elements of the matrix product as

$$\cos Az = \frac{\sin \delta \cos \phi - \cos \delta \cos t \sin \phi}{\cos h}$$

The ratio of the sine and cosine functions forms the following tangent function:

$$\tan Az = \frac{\cos \delta \sin t}{\sin \delta \cos \phi - \cos \delta \cos t \sin \phi}$$

### 2.4.2 Direction Cosine Matrix Rates

The time rate of change of the DCM is important to the later development of navigation state equations. This matrix differential equation governs the evolution of the corresponding DCM. This equation will be developed for the simple single-axis rotation presented in Fig. 2.1, then generalized for an arbitrary three-axis rotation.

Consider the incremental rotation  $\Delta\psi$  over the time interval  $\Delta t$  of the  $y$  frame shown in Fig. 2.7. Examining the moving  $y$ -frame components of the vector  $\underline{r}$  in Fig. 2.7, the future time  $t + \Delta t$  components can be expressed in terms of the current time  $t$  components as

$$y_1(t + \Delta t) = y_1(t) + \Delta\psi y_2(t) \quad (2.65)$$

$$y_2(t + \Delta t) = y_2(t) - \Delta\psi y_1(t) \quad (2.66)$$

or, in vector/matrix form,

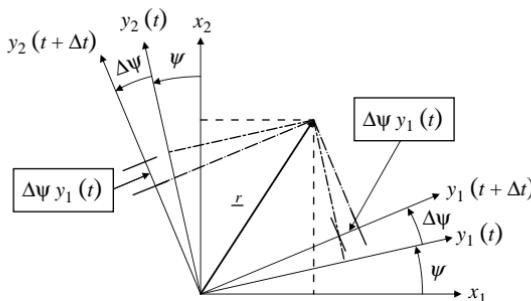
$$\begin{bmatrix} y_1(t + \Delta t) \\ y_2(t + \Delta t) \end{bmatrix} = \begin{bmatrix} 1 & \Delta\psi \\ -\Delta\psi & 1 \end{bmatrix} \begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} \quad (2.67)$$

Define

$$\begin{bmatrix} 1 & \Delta\psi \\ -\Delta\psi & 1 \end{bmatrix} \equiv I - \Delta\Psi \quad (2.68)$$

where

$$\Delta\Psi \equiv \begin{bmatrix} 0 & -\Delta\psi \\ \Delta\psi & 0 \end{bmatrix} \quad (2.69)$$



**Fig. 2.7 Rotating vector/coordinate frame components.**

At time  $t$ , the components along the  $y$  frame can be expressed in terms of the  $x$  frame components by the  $C$  matrix determined earlier. This is expressed by

$$\mathbf{y}(t) = C(t)\mathbf{x} \quad (2.70)$$

In terms of the  $x$  frame components, the values of  $y$  at the current time  $t + \Delta t$  are expressed by

$$\begin{aligned}\mathbf{y}(t + \Delta t) &= [I - \Delta\Psi]C(t)\mathbf{x} \\ &= C(t + \Delta t)\mathbf{x}\end{aligned} \quad (2.71)$$

The time rate of change of the  $C$  matrix is defined as

$$\begin{aligned}\dot{C}(t) &= \lim_{\Delta t \rightarrow 0} \frac{C(t + \Delta t) - C(t)}{\Delta t} \\ &= \lim_{\Delta t \rightarrow 0} \frac{C(t) - \Delta\Psi C(t) - C(t)}{\Delta t} \\ &= \lim_{\Delta t \rightarrow 0} -\frac{\Delta\Psi}{\Delta t} C(t)\end{aligned} \quad (2.72)$$

or

$$\dot{C}_x^y(t) = -\Omega_{x/y}^y(t)C_x^y(t) \quad (2.73)$$

where the skew-symmetric rotation matrix  $\Omega$  is

$$\Omega_{x/y}^y = \begin{bmatrix} 0 & -\omega_3 \\ \omega_3 & 0 \end{bmatrix} \quad (2.74)$$

The rotation matrix  $\Omega$  subscripts and superscript are interpreted as a rotation of the  $y$  frame relative to the  $x$  frame referenced in the  $y$  frame.

The transposed equivalent is

$$\begin{aligned}\dot{C}_y^x &= -C_y^x(\Omega_{x/y}^y)^T \\ &= C_y^x\Omega_{x/y}^y\end{aligned}$$

The second of these expressions is the result of the skew-symmetric form of the  $\Omega$  matrix. (Its transpose is the negative of the original matrix.) Expressing this final result for a three-axis rotation,

$$\dot{C}_y^x = C_y^x\Omega_{x/y}^y \quad (2.75)$$

where

$$\Omega_{x/y}^y = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (2.76)$$

and the rotations  $\omega_i$  ( $i = 1, 2$ , and  $3$ ) correspond to rotations about the  $x$ ,  $y$ , and  $z$  axes, respectively, of the  $y$  frame with respect to the  $x$  frame referenced in the  $y$  frame and written in vector notational form as  $\boldsymbol{\omega}_{x/y}^y$ . The skew-symmetric matrix in Eq. (2.76) is equivalent to the angular velocity vector's cross product  $(\boldsymbol{\omega}_{x/y}^y \times)$ .

The skew-symmetric rotation matrix in Eq. (2.76), referenced in one frame, can be related to a rotation matrix's rates referenced in another frame by using the similarity transformation

$$\Omega_{x/y}^y = C_x^y \Omega_{x/y}^x C_y^x \quad (2.77)$$

Note that the sense of the rotation, that is,  $y$  relative to  $x$ , is unchanged. The sense of rotation is reversed by changing the order of the subscripts of this matrix, accompanied with a corresponding change in sign of the matrix.

## 2.5 Miscellaneous Mathematical Topics

### 2.5.1 Angular Velocity Addition Theorem

Use is made later of the angular velocity addition theorem. This theorem states that for angular velocity vectors referenced in a common frame,

$$\boldsymbol{\omega}_{a/z} = \boldsymbol{\omega}_{a/b} + \boldsymbol{\omega}_{b/c} + \cdots + \boldsymbol{\omega}_{y/z} \quad (2.78)$$

### 2.5.2 Other Direction Cosine Matrix Dynamical Descriptions

The earlier DCM can be expressed in terms of an orientation unit vector  $\boldsymbol{\lambda}$  and a rotation angle  $\delta$  about this vector as (see Fig. 2.8)

$$C_y^x = \cos \delta I + \sin \delta (\boldsymbol{\lambda} \times) + (1 - \cos \delta) \boldsymbol{\lambda} \boldsymbol{\lambda}^T \quad (2.79)$$

Using the following vector relationship,

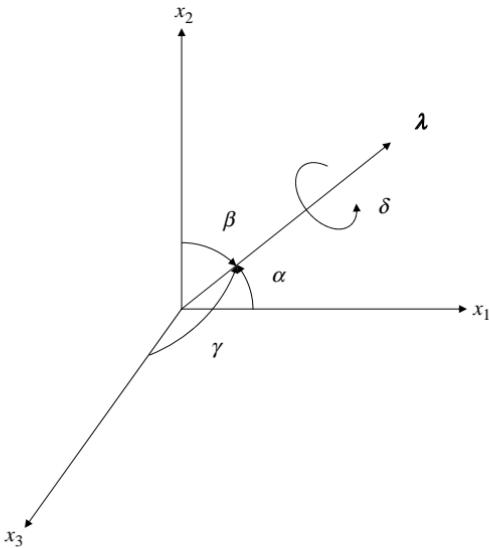
$$\boldsymbol{\lambda} \boldsymbol{\lambda}^T = I + (\boldsymbol{\lambda} \times)(\boldsymbol{\lambda} \times) \quad (2.80)$$

This matrix can also be written as

$$C_y^x = I + \sin \delta (\boldsymbol{\lambda} \times) + (1 - \cos \delta) (\boldsymbol{\lambda} \times)(\boldsymbol{\lambda} \times) \quad (2.81)$$

Quaternions are a four-parameter representation of a transformation matrix and are defined as follows:

$$q_0 = \cos \frac{\delta}{2} \quad (2.82)$$



**Fig. 2.8 Orientation and rotation angles.**

$$q_1 = \lambda_1 \sin \frac{\delta}{2} = \cos \alpha \sin \frac{\delta}{2} \quad (2.83)$$

$$q_2 = \lambda_2 \sin \frac{\delta}{2} = \cos \beta \sin \frac{\delta}{2} \quad (2.84)$$

$$q_3 = \lambda_3 \sin \frac{\delta}{2} = \cos \gamma \sin \frac{\delta}{2} \quad (2.85)$$

where

$$\cos^2 \alpha + \cos^2 \beta + \cos^2 \gamma = 1 \quad (2.86)$$

The angles  $\alpha$ ,  $\beta$ , and  $\gamma$  define the orientation of a unit vector defined with respect to the coordinate axes, as shown in Fig. 2.8.

The preceding DCM can be expressed using quaternions as

$$C_y^x = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (2.87)$$

where  $q_i$  satisfies the following orthogonality condition:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (2.88)$$

The dynamics of  $q_i$  are described by the following vector differential equation:

$$\frac{d}{dt} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_0 \end{bmatrix} = -\frac{1}{2} \begin{bmatrix} 0 & -\omega_3 & \omega_2 & -\omega_1 \\ \omega_3 & 0 & -\omega_1 & -\omega_2 \\ -\omega_2 & \omega_1 & 0 & -\omega_3 \\ \omega_1 & \omega_2 & \omega_3 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_0 \end{bmatrix} \quad (2.89)$$

Initialization values for  $q_i$  can be obtained from DCM elements.

Advantages for using quaternions include a reduction of dynamic variables, that is, from six for the direction cosine matrix to four for quaternions, and the elimination of possible singularities for three-variable Euler angle equations (see Chapter 5 problems). Disadvantages include nonlinear terms in the result and the necessity of renormalizations within the computational cycles to maintain properties already discussed for DCMs.

## 2.6 Chapter Summary

Vector/matrix algebra and calculus topics were reviewed in this chapter. These operations have been applied to the development of a least-squares estimate. An extension of this approach to weighted least squares (see Problem 2.4) will be used to develop the Kalman filter algorithm in Chapter 8. The linearization of nonlinear equations to obtain linearized forms of these equations was also illustrated. These techniques are required to linearize nonlinear navigation state equations, developed in Chapter 5, to obtain state error equations for implementation in the Kalman filter algorithm.

Transformations of vector components between different coordinate systems using DCMs were presented. Sequential rotations relative to intermediate coordinate frames were used to establish a transformation matrix relating vector components in one coordinate system to the components of that vector in another coordinate system, with the two coordinate systems being of arbitrary orientation to each other. The dynamic equations that describe the evolution of direction cosine matrices were developed. Direction cosine matrix dynamic equations are used in Chapter 5 to describe the attitude and position dynamics portion of navigation state equations (see Sec. 5.2).

Several miscellaneous mathematical topics, including the angular velocity addition theorem and a quaternion description for the DCM, were presented without development. The angular velocity addition theorem is used extensively in later developments of navigation state equations.

## Problems

- 2.1** Consider the problem of estimating an unknown constant from data that contain additive error. Many samples of data are available. Available data are related to the unknown constant as

$$z_i = a + v_i$$

where  $a = \text{constant}$  to be estimated and  $v = \text{zero mean additive error}$  with each sample. Using the least-squares estimate formed earlier in Example 2.3, find an estimate of the unknown constant  $a$ .

First, assume two samples are available from which to form an estimate. The problem can be reformulated as a vector as

$$\mathbf{z} = \mathbf{h}a + \mathbf{v}$$

or

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}a + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

Assume that additive errors are independent and uncorrelated, that is,

$$\mathbf{R} = \begin{bmatrix} r_1 & 0 \\ 0 & r_2 \end{bmatrix}$$

Using definitions for the measurement matrix  $H$  and unknown vector  $x$ , obtain the following expressions for the estimate and estimation error variance:

$$\hat{a} = \frac{1}{2}(z_1 + z_2)$$

$$p = \frac{1}{4}(r_1 + r_2)$$

The least-squares estimate of the constant with zero mean additive error is just the average of the data. Extend this result for the case of three measurements to obtain the following:

$$\hat{a} = \frac{1}{3}(z_1 + z_2 + z_3)$$

$$p = \frac{1}{9}(r_1 + r_2 + r_3)$$

Generalize this result to show that, in the general case of  $m$  measurements, the estimate of the constant and the estimation error variance becomes

$$\hat{a} = \frac{1}{m} \sum_{i=1}^m z_i$$

$$p = \frac{1}{m^2} \sum_{i=1}^m r_i$$

And, if additive errors are identically distributed with variance  $r$ , the estimation error variance becomes

$$p = \frac{1}{m} r$$

This result shows that the estimate of the constant is the mean or average of the data samples and that the uncertainty associated with the estimate is reduced by the number of samples used to determine the estimate.

- 2.2** Consider the problem of estimating radial position error rate circular error probable (CEP) from time-tagged radial position errors. From Example 2.3, an estimate of the slope  $m$  is desired where the intercept  $b$  is defined as zero. The measurement matrix (vector) for  $m$  observations becomes

$$\mathbf{h} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ \vdots \\ t_m \end{bmatrix}$$

for  $m$  radial position error observations

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_m \end{bmatrix}$$

Show that the least-squares estimate for slope  $m$  can be expressed as

$$\hat{x} = \frac{\sum_{i=1}^m z_i h_i}{\sum_{i=1}^m h_i^2} = \hat{m} = \frac{\sum_{i=1}^m z_i t_i}{\sum_{i=1}^m t_i^2}$$

where the subscripts correspond to vector components.

To establish the CEP, a number of sample sets are used and an estimate of each set's slope computed. For  $N$  test sample sets, if errors associated with the radial error rate are Gaussian, then the CEP is computed by

$$\text{CEP}_{\text{RER}} = 0.83 \sqrt{\sum_{i=1}^N \hat{m}_i^2 / N}$$

where RER is radial error rate.

- 2.3** Gyro static test data are analyzed to characterize statistically independent error contributors. This exercise applies the least-squares methodology to estimate these error contributors. Data groups are formed corresponding to different time spans  $\tau$ , and corresponding variances are computed  $\sigma^2(\tau)$ . The model defined for these variances (square-root) is

$$\sigma(t) = a_0 + \frac{a_{-1/2}}{\sqrt{\tau}} + \frac{a_{-1}}{\tau}$$

The  $a$ 's in this equation are unknown and are to be determined. Assume a vector defined in terms of these parameters as

$$\mathbf{x} = \begin{bmatrix} a_0 \\ a_{-1/2} \\ a_{-1} \end{bmatrix}$$

Show that the  $H$  matrix row (vector) for each observation is

$$\mathbf{h}_i = \begin{bmatrix} 1 & \frac{1}{\sqrt{\tau_i}} & \frac{1}{\tau_i} \end{bmatrix}^T$$

The variances so computed are known as Allan variances, and the coefficients of the correspond to the following error contributors:

Quantization:

$$\sigma_Q^2(\tau) \sim f_Q\left(\frac{1}{\tau^2}\right)$$

Angle random walk:

$$\sigma_{rw}^2(\tau) \sim f_{rw}\left(\frac{1}{\tau}\right)$$

and

Bias stability:

$$\sigma_b^2(\tau) \sim f_b\left(\frac{1}{\tau^0}\right)$$

- 2.4** This exercise develops a weighted least-squares estimate, extending Example 2.3. The weighted least-squares cost function form is used in the development of the Kalman filter algorithm. The cost function is modified by weighting the vector inner product with a symmetric matrix  $W$  as follows:

$$J_{WLS} = (\mathbf{z} - H\mathbf{x})^T W (\mathbf{z} - H\mathbf{x})$$

Expand terms to yield

$$\begin{aligned} J_{\text{WLS}} &= \mathbf{z}^T W \mathbf{z} - \mathbf{x}^T H^T W \mathbf{z} - \mathbf{z}^T W H \mathbf{x} + \mathbf{x}^T H^T W H \mathbf{x} \\ &= \alpha - \mathbf{x}^T \mathbf{y}' - \mathbf{y}''^T \mathbf{x} + \mathbf{x}^T (\mathbf{A} \mathbf{x}) \end{aligned}$$

where

$\alpha$  = scalar (no functional dependence on  $\mathbf{x}$ )

$\mathbf{y}' = H^T W \mathbf{z} = (n \times 1)$  column vector

$\mathbf{y}''^T = \mathbf{z}^T W H = (1 \times n)$  row vector

$\mathbf{A} = H^T W H = (n \times n)$  matrix

Recall Eqs. (2.24) and (2.25):

$$a = \mathbf{x}^T \mathbf{y} = \mathbf{y}^T \mathbf{x}$$

$$\frac{\partial a}{\partial \mathbf{x}} = \mathbf{y}^T$$

With the following expression for the vector  $\mathbf{y}$

$$\mathbf{y} = A \mathbf{x}$$

and, from Eq. (2.26), its derivative is

$$\frac{\partial \mathbf{y}}{\partial \mathbf{x}} = A$$

These definitions are used in the minimization of the cost function  $J$  with respect to the vector  $\mathbf{x}$ .

Take the partial derivative of each term in the preceding expanded equation to yield

$$\begin{aligned} \frac{\partial J_{\text{WLS}}}{\partial \mathbf{x}} &= -\mathbf{y}''^T - \mathbf{y}''^T + \frac{\partial}{\partial \mathbf{x}} (\mathbf{x}^T \mathbf{y}) \\ &= -\mathbf{z}^T W H - \mathbf{z}^T W H + \mathbf{x}^T H^T W H + \mathbf{x}^T H^T W H \\ &= -2(\mathbf{z}^T W H + \mathbf{x}^T H^T W H) \end{aligned}$$

Setting this result to zero, transposing, and solving for the vector  $\mathbf{x}$ ,

$$(\mathbf{x}^T H^T W H)^T = (\mathbf{z}^T W H)^T \implies (H^T W H) \mathbf{x} = (W H) \mathbf{z}$$

or

$$\hat{\mathbf{x}} = (H^T W H)^{-1} H^T W \mathbf{z}$$

The following properties are to be developed as in Example 2.3, replacing the weighting matrix  $W$  with the matrix  $R^{-1}$ .

1) Estimation error:

$$\begin{aligned}\boldsymbol{\varepsilon} &\equiv \mathbf{x} - \hat{\mathbf{x}} \\ &= \mathbf{x} - (H^T R^{-1} H)^{-1} H^T R^{-1} \mathbf{z} \\ &= \mathbf{x} - (H^T R^{-1} H)^{-1} H^T R^{-1} (H\mathbf{x} + \mathbf{v}) \\ &= \mathbf{x} - (H^T R^{-1} H)^{-1} (H^T R^{-1} H)\mathbf{x} - (H^T R^{-1} H)^{-1} H^T R^{-1} \mathbf{v} \\ &= -(H^T R^{-1} H)^{-1} H^T R^{-1} \mathbf{v}\end{aligned}$$

If the errors are random zero mean processes such that

$$E[\mathbf{v}] \equiv 0 \quad \text{and} \quad E[\mathbf{v} \mathbf{v}^T] \equiv R$$

then

2) Estimation error covariance:

$$\begin{aligned}E[(\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T] &= E\{[(H^T R^{-1} H)^{-1} H^T R^{-1} \mathbf{v}][(H^T R^{-1} H)^{-1} H^T R^{-1} \mathbf{v}]^T\} \\ &= (H^T R^{-1} H)^{-1} H^T R^{-1} E[\mathbf{v} \mathbf{v}^T] R^{-1} H (H^T R^{-1} H)^{-1} \\ &= (H^T R^{-1} H)^{-1} H^T R^{-1} R R^{-1} H (H^T R^{-1} H)^{-1} \\ &= (H^T R^{-1} H)^{-1} (H^T R^{-1} H) (H^T R^{-1} H)^{-1} \\ &= (H^T R^{-1} H)^{-1} \\ &\equiv P\end{aligned}$$

- 2.5** This exercise exploits the orthonormal properties of direction cosine matrices. Consider the following form for a DCM:

$$C \equiv [\mathbf{u} | \mathbf{v} | \mathbf{w}]$$

where its columns are orthogonal unit vectors. With this property,

$$\mathbf{u} \times \mathbf{v} = \mathbf{w}$$

$$\mathbf{v} \times \mathbf{w} = \mathbf{u}$$

and

$$\mathbf{w} \times \mathbf{u} = \mathbf{v}$$

From these cross products, the third element of each can be obtained as

$$\begin{aligned} u_1 v_2 - u_2 v_1 &= w_3 \\ v_1 w_2 - v_2 w_1 &= u_3 \end{aligned}$$

and

$$w_1 u_2 - w_2 u_1 = v_3$$

Returning to the definition of the preceding DCM, the following are obtained for the DCM's third "row" in terms of the other rows:

$$\begin{aligned} C_{11}C_{22} - C_{21}C_{12} &= C_{33} \\ C_{12}C_{23} - C_{22}C_{13} &= C_{31} \end{aligned}$$

and

$$C_{13}C_{21} - C_{23}C_{11} = C_{32}$$

These last three equations are useful in relating three of the DCM's elements to its others. This property is beneficial because it allows a numerical integration algorithm that implements only two of the three rows of the DCM, integrating six rather than nine differential equations.

- 2.6** Additional terms in state transition matrix are developed in this exercise. Continue the expansion of the perturbation state in a Taylor-series beyond the first term

$$\delta\mathbf{x}(t + \Delta t) = \delta\mathbf{x}(t) + \frac{\partial \delta\mathbf{x}(t)}{\partial t} \Delta t + \frac{1}{2} \frac{\partial^2 \delta\mathbf{x}(t)}{\partial t^2} \Delta t^2 + \dots$$

where  $\partial \delta\mathbf{x}/\partial t \equiv \dot{\delta}\mathbf{x}$ , etc. Then, by interchanging the order of differentiation, obtain the following

$$\frac{\partial^2 \delta\mathbf{x}}{\partial t^2} = \frac{\partial}{\partial t} (\dot{\delta}\mathbf{x}) = \frac{\partial}{\partial t} \left( \frac{\partial f}{\partial \mathbf{x}} \delta\mathbf{x} \right) \cong \frac{\partial f}{\partial \mathbf{x}} \frac{\partial \delta\mathbf{x}}{\partial t} = \frac{\partial f}{\partial \mathbf{x}} \frac{\partial f}{\partial \mathbf{x}} \delta\mathbf{x}$$

assuming that  $\partial f/\partial \mathbf{x}|_{x=x_0} \equiv F$  is a constant matrix. Substitute this expression, using the  $F$  notation, into the preceding expansion, to obtain

the following:

$$\begin{aligned}\delta\mathbf{x}(t + \Delta t) &= \delta\mathbf{x}(t) + F\Delta t\delta\mathbf{x} + \frac{1}{2}F^2\Delta t^2\delta\mathbf{x} + \dots \\ &= [I + F\Delta t + \frac{1}{2}F^2\Delta t^2 + \dots]\delta\mathbf{x}(t) \\ &\equiv \Phi_{t,t+\Delta t}\delta\mathbf{x}(t)\end{aligned}$$

Define the state transition matrix as

$$\Phi_{t,t+\Delta t} \equiv [I + F\Delta t + \frac{1}{2}F^2\Delta t^2 + \dots] \implies e^{F\Delta t}$$

- 2.7** Obtain a linear equation relating the change in range to perturbations about nominal position. The three-dimensional position components  $\Delta x$ ,  $\Delta y$ , and  $\Delta z$  are used to form range as  $\rho = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$ . Defining the nominal relative position vector as

$$\Delta\mathbf{x} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}$$

and its perturbation as

$$\delta\mathbf{x} = \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}$$

then use the truncated Taylor-series expansion to evaluate the perturbed range as  $\rho(\Delta\mathbf{x} + \delta\mathbf{x}) \approx \rho(\Delta\mathbf{x}) + (\partial\rho/\partial\Delta\mathbf{x})\delta\mathbf{x}$  to obtain the range error  $\delta\rho = \rho(\Delta\mathbf{x} + \delta\mathbf{x}) - \rho(\Delta\mathbf{x})$  related to perturbed position vector as

$$\delta\rho \approx \left[ \frac{\Delta x}{\rho} \quad \frac{\Delta y}{\rho} \quad \frac{\Delta z}{\rho} \right] \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}$$

- 2.8** In this exercise, a numerical algorithm solution to the DCM differential equation is developed. The homogeneous matrix differential equation

$$\dot{X}(t) = A(t)X(t)$$

has the solution

$$X(t) = \Phi(t, t_0)X(t_0)$$

where the state transition matrix can be expressed in terms of the matrix exponential

$$\begin{aligned}\Phi(t, t_0) &= \exp \left[ \int_{t_0}^t A(t) dt \right] \\ &\approx e^{A\Delta t} \\ &\approx \left( I + A\Delta t + A^2 \frac{\Delta t^2}{2!} + A^3 \frac{\Delta t^3}{3!} + A^4 \frac{\Delta t^4}{4!} \right)\end{aligned}$$

for small  $\Delta t = t - t_0$  and constant  $A$ .

Consider the following form of the DCM differential equation:

$$\dot{C}_x^y = -\Omega_{x/y}^y C_x^y$$

The skew-symmetric rotation matrix corresponds to the preceding matrix  $A$ . Then, the product  $A \Delta t$  is written as

$$A\Delta t = \begin{bmatrix} 0 & \Delta\theta_3 & -\Delta\theta_2 \\ -\Delta\theta_3 & 0 & \Delta\theta_1 \\ \Delta\theta_2 & -\Delta\theta_1 & 0 \end{bmatrix}$$

Show that, up to the fourth-order term, the approximation to the matrix exponential can be expressed as

$$e^{A\Delta t} \approx \begin{bmatrix} 1 - b(\Delta\theta_2^2 + \Delta\theta_3^2) & a\Delta\theta_3 + b\Delta\theta_1\Delta\theta_2 & -a\Delta\theta_2 + b\Delta\theta_1\Delta\theta_3 \\ -a\Delta\theta_3 + b\Delta\theta_1\Delta\theta_2 & 1 - b(\Delta\theta_1^2 + \Delta\theta_3^2) & a\Delta\theta_1 + b\Delta\theta_2\Delta\theta_3 \\ a\Delta\theta_2 + b\Delta\theta_1\Delta\theta_3 & -a\Delta\theta_1 + b\Delta\theta_2\Delta\theta_3 & 1 - b(\Delta\theta_1^2 + \Delta\theta_2^2) \end{bmatrix}$$

where

$$a \equiv 1 - \sum_{i=1}^3 \Delta\theta_i^2 / 6$$

and

$$b \equiv \frac{1}{2} - \sum_{i=1}^3 \Delta\theta_i^2 / 24$$

Once initialized, the DCM is integrated between time steps using this expression.

- 2.9** A numerical algorithm solution for a nonhomogeneous vector differential equation is developed in this exercise. The vector differential equation

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + \mathbf{u}$$

has the solution

$$\mathbf{x}(t) = e^{A\Delta t}\mathbf{x}(t_0) + A^{-1}(e^{A\Delta t} - 1)\mathbf{u}$$

where

$$A^{-1}(e^{A\Delta t} - 1) \approx \left( I + A \frac{\Delta t}{2} + A^2 \frac{\Delta t^2}{3!} + A^3 \frac{\Delta t^3}{4!} \right) \Delta t$$

up to the fourth-order.

Using the definition of the matrix  $A$  from Problem 2.8, show that this equation can be expressed as

$$A^{-1}(e^{A\Delta t} - I) \approx \begin{bmatrix} a + c\Delta\theta_1^2 & b\Delta\theta_3 + c\Delta\theta_1\Delta\theta_2 & -b\Delta\theta_2 + c\Delta\theta_1\Delta\theta_3 \\ -b\Delta\theta_3 + c\Delta\theta_1\Delta\theta_2 & a + c\Delta\theta_2^2 & b\Delta\theta_1 + c\Delta\theta_2\Delta\theta_3 \\ b\Delta\theta_2 + c\Delta\theta_1\Delta\theta_3 & -b\Delta\theta_1 + c\Delta\theta_2\Delta\theta_3 & a + c\Delta\theta_3^2 \end{bmatrix} \Delta t$$

where  $b$  was given earlier and  $c = 1/6$ .

- 2.10** Solve for quaternions in terms of elements of the  $C_y^x$  matrix. The trace of this matrix is

$$\begin{aligned} \text{tr}[C_y^x] &\equiv C_{y11}^x + C_{y22}^x + C_{y33}^x \\ &= 4q_0^2 - 1 \end{aligned}$$

Obtain the following:

$$\begin{aligned} q_1 &= \frac{C_{y32}^x - C_{y23}^x}{4q_0} \\ q_2 &= \frac{C_{y13}^x - C_{y31}^x}{4q_0} \\ q_3 &= \frac{C_{y21}^x - C_{y12}^x}{4q_0} \end{aligned}$$

These expressions provide quaternion values, from DCM elements, to initialize the quaternion differential equation (2.89).

- 2.11** This exercise reformulates the quaternion dynamics in Eq. (2.89) and establishes several quaternion relationships to be used later in the Attitude Determination and Estimation chapter. Rewrite Eq. (2.89) as

$$\dot{\mathbf{q}} = -\frac{1}{2} Z^y \boldsymbol{\omega}$$

where the matrix  $Z^y$  is defined as

$$Z^y \equiv \begin{bmatrix} -q_0 & q_3 & -q_2 \\ -q_3 & -q_0 & q_1 \\ q_2 & -q_1 & -q_0 \\ q_1 & q_2 & q_3 \end{bmatrix}$$

Define another matrix  $Z^x$  as

$$Z^x \equiv \begin{bmatrix} -q_0 & -q_3 & q_2 \\ q_3 & -q_0 & -q_1 \\ -q_2 & q_1 & -q_0 \\ q_1 & q_2 & q_3 \end{bmatrix}$$

and show the following relationships:

$$C_y^x = Z^{x^T} Z^y, \quad Z^x C_y^x = Z^y, \quad Z^{y^T} \mathbf{q} = 0_{3 \times 1} \quad \text{and} \quad Z^{y^T} Z^y = Z^{x^T} Z^x = I_{3 \times 3}$$

- 2.12** Confirm the similarity transformation in Eq. (2.77) in the following. Consider the following rotation vector change in coordinate (transformation) from the  $x$  frame to the  $y$  frame, where the parentheses  $( )$  is the place where the rotation to and from frames is usually located.

$$\begin{aligned} \boldsymbol{\omega}_{()}^y &= C_x^y \boldsymbol{\omega}_{()}^x \\ &= \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} \omega_1^x \\ \omega_2^x \\ \omega_3^x \end{bmatrix} \\ &= \begin{bmatrix} C_{11}\omega_1^x + C_{12}\omega_2^x + C_{13}\omega_3^x \\ C_{21}\omega_1^x + C_{22}\omega_2^x + C_{23}\omega_3^x \\ C_{31}\omega_1^x + C_{32}\omega_2^x + C_{33}\omega_3^x \end{bmatrix} \end{aligned}$$

The similarity transformation is defined as

$$\begin{aligned}\Omega_{()^y}^y &= C_x^y \Omega_{()^x}^x C_y^x \\ &= \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} 0 & -\omega_3^x & \omega_2^x \\ \omega_3^x & 0 & -\omega_1^x \\ -\omega_2^x & \omega_1^x & 0 \end{bmatrix} \begin{bmatrix} C_{11} & C_{21} & C_{31} \\ C_{12} & C_{22} & C_{32} \\ C_{13} & C_{23} & C_{33} \end{bmatrix}\end{aligned}$$

Equating the 3:2 elements of this product

$$\begin{aligned}\Omega_{(32)}^y &= \omega_1^y = C_{31}(-\omega_3^x C_{22} + \omega_2^x C_{23}) + C_{32}(\omega_3^x C_{21} - \omega_1^x C_{23}) \\ &\quad + C_{33}(-\omega_2^x C_{21} + \omega_1^x C_{22}) \\ &= (C_{22}C_{33} - C_{32}C_{23})\omega_1^x + (C_{31}C_{23} - C_{33}C_{21})\omega_2^x \\ &\quad + (C_{32}C_{21} - C_{31}C_{22})\omega_3^x \\ &= C_{11}\omega_1^x + C_{12}\omega_2^x + C_{13}\omega_3^x\end{aligned}$$

Because

$$(C_{22}C_{33} - C_{32}C_{23}) = C_{11}, \text{ etc.}$$

Equate the two other elements to complete the demonstration of the similarity transformation.

- 2.13** This is an advanced exercise for the development of an iterative algorithm for orthonormalization of a DCM. It is desired to minimize the deviation of the computed DCM and the true DCM ( $\bar{C} - C$ ) subject to the constraint that the true DCM satisfies

$$C^T C = I$$

Consider the following scalar cost function:

$$J = \text{tr}[(\bar{C} - C)^T (\bar{C} - C) + \Lambda(C^T C - I)]$$

where the preceding constraint has been included in the cost function via the Lagrange multiplier matrix  $\Lambda$  and  $\Lambda$  is assumed to be a symmetric matrix. Minimizing this function (taking the partial derivative with respect to  $C$  and equating the result to the null matrix) yields the following:

$$0_{3 \times 3} = 2[-\bar{C}^T + (I + \Lambda)C^T]$$

The value of  $C$  that minimizes the cost function becomes

$$\hat{C} = \bar{C}(I + \Lambda)^{-1}$$

where it is assumed that the matrix  $(I + \Lambda)$  is nonsingular.

Next, the preceding constraint equation is used to eliminate  $(I + \Lambda)$  in the result just obtained. Substituting that result into the

constraint equation,

$$\hat{C}^T \hat{C} = I = (I + \Lambda)^{-1} \bar{C}^T \bar{C} (I + \Lambda)^{-1}$$

and solving for  $(I + \Lambda)$  yields

$$I + \Lambda = \pm (\bar{C}^T \bar{C})^{1/2}$$

Substituting this expression into the preceding  $\hat{C}$  equation yields

$$\hat{C} = \pm \bar{C} (\bar{C}^T \bar{C})^{1/2}$$

An iterative form of this equation can be obtained by first forming the error matrix

$$E \equiv \bar{C}^T \bar{C} - I$$

or

$$\bar{C}^T \bar{C} = I + E$$

then

$$\hat{C} = \pm \bar{C} (I + E)^{1/2}$$

If the error matrix  $E$  is sufficiently small, then  $(I + E)^{-1/2}$  can be expressed in the series form

$$(I + E)^{-1/2} \approx I - \frac{1}{2}E + \frac{1*3}{2*4}E^2 - \mathcal{O}(E^3)$$

Retaining only the first two terms in this series and substituting into the second  $\hat{C}$  equation yields

$$\begin{aligned}\hat{C} &\approx \bar{C} \left( I - \frac{1}{2}E \right) \\ &= \bar{C} \left[ I - \frac{1}{2}(\bar{C}^T \bar{C} - I) \right] \\ &= \frac{3}{2} \bar{C} - \frac{1}{2} \bar{C} \bar{C}^T \bar{C}\end{aligned}$$

The iterative character of this equation can be established by using the following replacements:

$$\hat{C} \leftarrow C_{n+1}$$

and

$$\bar{C} \leftarrow C_n$$

Then, the iteration algorithm becomes

$$C_{n+1} = \frac{3}{2}C_n - \frac{1}{2}C_n C_n^T C_n$$

- 2.14** The inverse of a partitioned matrix is considered in this exercise. Consider the matrix

$$\Sigma = \begin{bmatrix} \sum_{xx} & \sum_{xy} \\ \sum_{yx} & \sum_{yy} \end{bmatrix}$$

where it is assumed that the matrices  $\Sigma$  and  $\Sigma_{yy}$  are nonsingular. Develop the following inverse of  $\Sigma$  in terms of its partitioned elements:

$$\begin{aligned} \Sigma^{-1} &= \begin{bmatrix} I & 0 \\ -\sum_{yy}^{-1} \sum_{xy}^T & I \end{bmatrix} \begin{bmatrix} (\sum_{xx} - \sum_{xy} \sum_{yy}^{-1} \sum_{yx})^{-1} & 0 \\ 0 & \sum_{yy}^{-1} \end{bmatrix} \\ &\quad \times \begin{bmatrix} I & -\sum_{xy} \sum_{yy}^{-1} \\ 0 & I \end{bmatrix} \end{aligned}$$

and the following expression for the determinate of  $\Sigma$

$$|\Sigma| = \left| \sum_{xx} - \sum_{xy} \sum_{yy}^{-1} \sum_{yx} \right| |\Sigma_{yy}|$$

Hint: Use the following transformation to transform this matrix into a diagonal form:

$$\begin{aligned} &\begin{bmatrix} I & -\sum_{xy} \sum_{yy}^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} \sum_{xx} & \sum_{xy} \\ \sum_{yx} & \sum_{yy} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\sum_{yy}^{-1} \sum_{xy}^T & I \end{bmatrix} \\ &= \begin{bmatrix} \sum_{xx} - \sum_{xy} \sum_{yy}^{-1} \sum_{yx} & 0 \\ \sum_{yx} & \sum_{yy} \end{bmatrix} \begin{bmatrix} I & 0 \\ -\sum_{yy}^{-1} \sum_{xy}^T & I \end{bmatrix} \\ &= \begin{bmatrix} \sum_{xx} - \sum_{xy} \sum_{yy}^{-1} \sum_{yx} & 0 \\ 0 & \sum_{yy} \end{bmatrix} \end{aligned}$$

- 2.15** A factor of the symmetric  $P$  matrix presented in Example 2.1 is considered again by developing a square-root factor defined by

$$P \equiv SS^T$$

where the columns of square-root matrix are written in terms of vectors as

$$S = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \cdots \quad \mathbf{v}_n]$$

Show that the  $P$  matrix can be expressed as

$$P = \sum_{i=1}^n \mathbf{v}_i \mathbf{v}_i^T$$

Hint: Establish the factors by considering the matrix  $P$  expressed in terms of a  $3 \times 3$  matrix.

$$\begin{aligned} P &= \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{12} & p_{22} & p_{23} \\ p_{13} & p_{23} & p_{33} \end{bmatrix} = \begin{bmatrix} s_{11} & 0 & 0 \\ s_{12} & s_{22} & 0 \\ s_{13} & s_{23} & s_{33} \end{bmatrix} \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ 0 & s_{22} & s_{23} \\ 0 & 0 & s_{33} \end{bmatrix} \\ &= \begin{bmatrix} s_{11}^2 & s_{11}s_{12} & s_{11}s_{13} \\ s_{11}s_{12} & s_{12}^2 + s_{22}^2 & s_{12}s_{13} + s_{22}s_{23} \\ s_{11}s_{13} & s_{12}s_{13} + s_{22}s_{23} & s_{13}^2 + s_{23}^2 + s_{33}^2 \end{bmatrix} \end{aligned}$$

Define the vectors and resulting outer product as

$$\begin{aligned} \mathbf{v}_1 &= \begin{bmatrix} s_{11} \\ s_{12} \\ s_{13} \end{bmatrix} \quad \mathbf{v}_1 \mathbf{v}_1^T = \begin{bmatrix} s_{11} \\ s_{12} \\ s_{13} \end{bmatrix} \begin{bmatrix} s_{11} & s_{12} & s_{13} \end{bmatrix} = \begin{bmatrix} s_{11}^2 & s_{11}s_{12} & s_{11}s_{13} \\ s_{11}s_{12} & s_{12}^2 & s_{12}s_{13} \\ s_{11}s_{13} & s_{12}s_{13} & s_{13}^2 \end{bmatrix} \\ \mathbf{v}_2 &= \begin{bmatrix} 0 \\ s_{22} \\ s_{23} \end{bmatrix} \quad \mathbf{v}_2 \mathbf{v}_2^T = \begin{bmatrix} 0 \\ s_{22} \\ s_{23} \end{bmatrix} \begin{bmatrix} 0 & s_{22} & s_{23} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & s_{22}^2 & s_{22}s_{23} \\ 0 & s_{22}s_{23} & s_{23}^2 \end{bmatrix} \\ \mathbf{v}_3 &= \begin{bmatrix} 0 \\ 0 \\ s_{33} \end{bmatrix} \quad \mathbf{v}_3 \mathbf{v}_3^T = \begin{bmatrix} 0 \\ 0 \\ s_{33} \end{bmatrix} \begin{bmatrix} 0 & 0 & s_{33} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & s_{33}^2 \end{bmatrix} \end{aligned}$$

Then, the  $P$  matrix becomes

$$P = \mathbf{v}_1\mathbf{v}_1^T + \mathbf{v}_2\mathbf{v}_2^T + \mathbf{v}_3\mathbf{v}_3^T$$

and the elements for these vectors become

$$\begin{aligned}s_{11}^2 &= p_{11} & s_{12} &= p_{12} / s_{11} & s_{13} &= p_{13} / s_{11} \\s_{22}^2 &= p_{22} - s_{12}^2 & s_{23} &= \frac{(p_{23} - s_{12}s_{13})}{s_{22}} \\s_{33}^2 &= p_{33} - (s_{13}^2 + s_{23}^2)\end{aligned}$$



## Coordinate Systems and Transformations

Navigation system states—position, velocity, and attitude—are defined with reference to coordinate frames, and there are a number of different frames in current use. Because there are differences, the survey of these systems presented in this chapter is intended to help reader understand information that is communicated using these frames.

This chapter presents a survey of coordinate systems and coordinate frame transformations. Coordinate systems that are currently used in navigation systems are presented first. These include several navigation frames, including Earth-centered, local geodetic, and wander azimuth. Certain local geodetic frames are usually associated with specific Earth-centered frames. These and their associations are presented. Body-axis frame definitions are also presented. Like the association between certain local geodetic frames and Earth-centered frames, various body frames are also associated with specific local geodetic frames. These associations are also presented. For the coordinate systems defined, transformation matrices (direction cosine matrices) are developed to transform navigation variables from one frame to another. Depending on the axis system definitions, different forms of the corresponding transformation matrices are obtained.

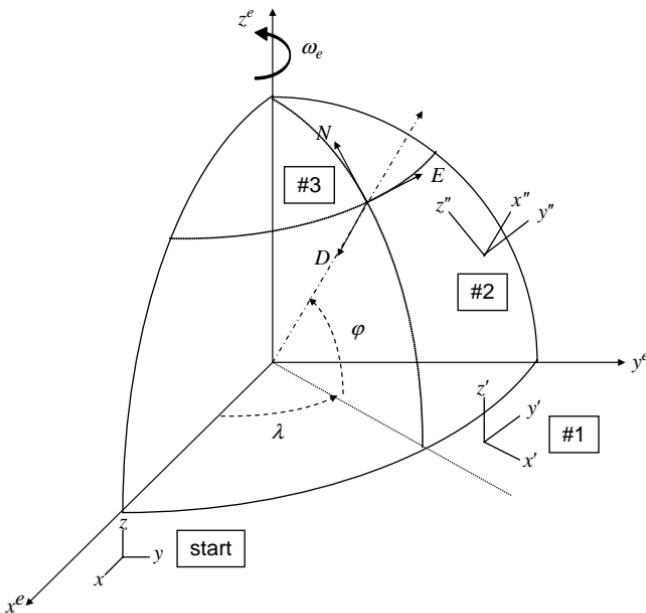
The concept of an external error vector associated with a direction cosine matrix is presented, and the relationship between this external error and errors in Euler angles is illustrated for position and attitude matrices.

### 3.1 Coordinate Systems

Coordinate systems are established to allow information to be exchanged between interfacing systems in a consistent manner. This section reviews the definitions of some commonly used coordinate systems.

#### 3.1.1 *Earth-Centered, Earth-Fixed Frame*

The Earth-centered, Earth-fixed (ECEF) frame is fixed within the Earth and its rotation, and it is centered at the Earth's center. Axis definitions in current use vary. Shown in Figs. 3.1, 3.2, and 3.3 are illustrations of three possible ECEF frames. In the first frame, shown in Fig. 3.1, the  $z$  axis is parallel to and aligned with the direction of the Earth's rotation. In the equatorial plane, the  $x$



**Fig. 3.1 ECEF coordinate frame with  $z$  axis along Earth's rotation axis.**

axis locates the Greenwich meridian, and the  $y$  axis completes the right-hand system. In Figs. 3.2 and 3.3, the direction of the Earth's rotation is parallel to the  $y$  and  $x$  axes, respectively. In Fig. 3.2, the  $z$  axis locates the Greenwich meridian, and the  $x$  axis completes the right-hand system. In the third frame, the  $z$  axis locates the Greenwich meridian, and the  $y$  axis completes the right-hand system.

Illustrated in these figures are intermediate frames resulting from single axis rotations. Sequences of rotations will be used to form the direction cosine matrices needed to transformation vector components from the ECEF frame to a local geodetic frame.

### 3.1.2 Earth-Centered Inertial Frame

In Figs. 3.1–3.3, a corresponding Earth-centered inertial (ECI) frame exists and is established by the direction of the Earth's rotation. This inertial frame is fixed to an inertial reference. The further definition of an inertial frame is not used in this treatment of navigation systems. If the reader is incorporating navigation aids that are based on stellar updates, then the inertial reference would have to be defined.

### 3.1.3 Local Geodetic Frame

Usually associated with the ECEF frames shown in Figs. 3.1, 3.2, and 3.3 are local geodetic (geographic) frames.

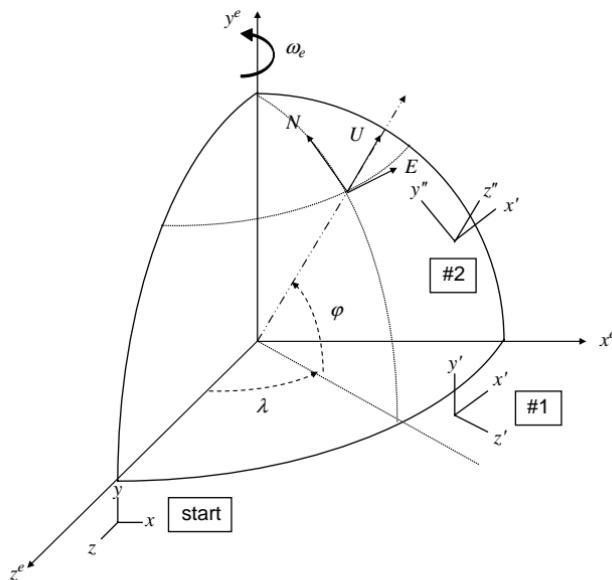


Fig. 3.2 ECEF coordinate frame with  $y$  axis along Earth's rotation axis.

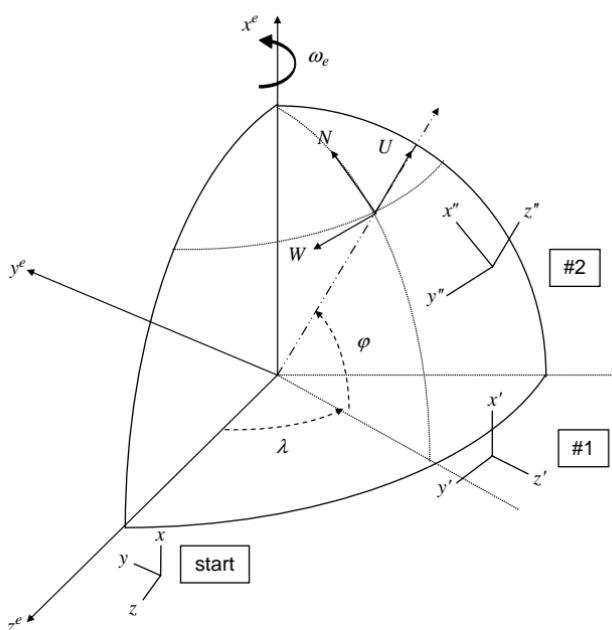


Fig. 3.3 ECEF coordinate frame with  $x$  axis along Earth's rotation axis.

### 3.1.4 Wander Azimuth Frame

The wander-azimuth (WA) frame is a local geodetic frame that is in current use for many navigation systems, and there are several WA frame definitions. The WA frames are usually associated with local geodetic frames shown in Figs. 3.1–3.3 and are the result of an additional rotation about the geodetic frame's vertical axis.

In Fig. 3.4, the WA frame is rotated with respect to the local geodetic frame  $z$  axis (down) by the WA angle  $\alpha$ . For the last two frames in Figs. 3.2 and 3.3, the corresponding WA frame and angle are defined as shown in Figs. 3.5 and 3.6. In these figures, the WA frame is rotated with respect to the local geodetic frame  $z$  axis (up) by the WA angle  $\alpha$ .

It will be shown later that the angle  $\alpha$ , whose rotation direction is about the  $z$  axis (down) axis (shown in Fig. 3.4), satisfies the following differential equation:

$$\dot{\alpha} = \lambda \sin \phi \quad (3.1)$$

whereas the corresponding equation for the other two WA frames, whose rotation direction is about the  $z$  axis (up) axis (shown in Figs. 3.5 and 3.6), is

$$\dot{\alpha} = -\lambda \sin \phi \quad (3.2)$$

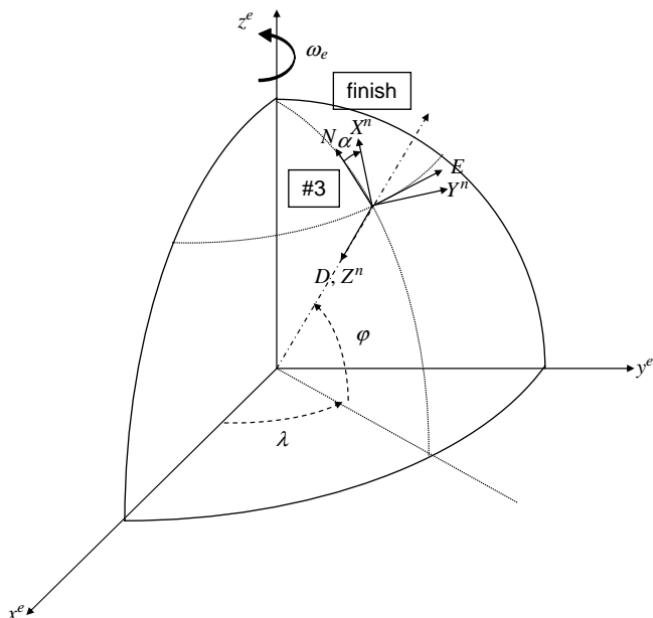
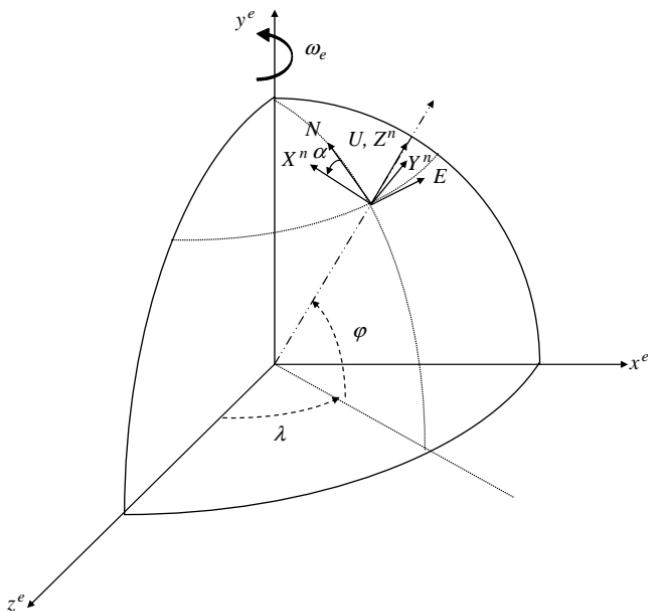
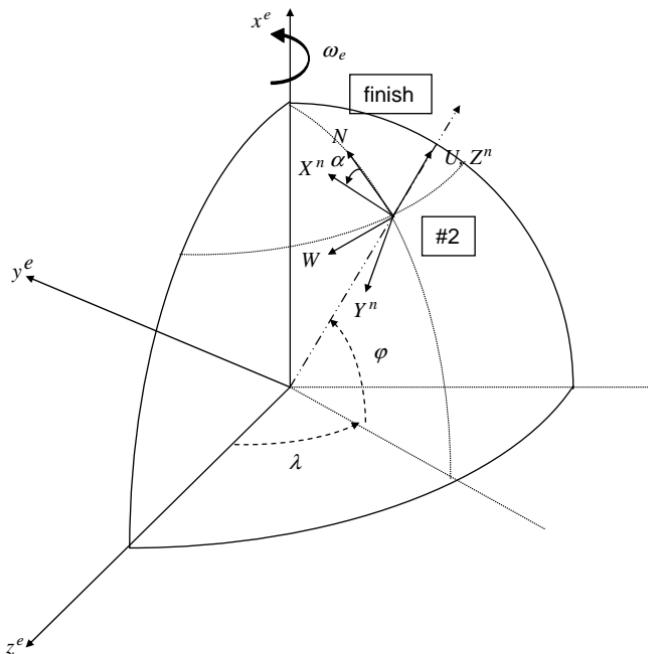


Fig. 3.4 Local-level WA frame for Fig. 3.1.



**Fig. 3.5** Wander azimuth for Fig. 3.2.



**Fig. 3.6** Wander azimuth for Fig. 3.3.

### 3.1.5 Body Frame

The body frame is rigidly attached to and defined within the vehicle carrying the navigation system. For strap-down navigation units, the body frame is usually identical to, or the result of a fixed transformation relative to, the navigation system inertial sensor's case frame.

Conventional definitions for this frame have the  $x$  axis along the vehicle longitudinal axis, the  $z$  axis downward, and the  $y$  axis pointed outward, completing the right-hand set. This frame is shown in Fig. 3.7 with yaw  $\Psi$ , pitch  $\theta$ , and roll  $\phi$  rotations about navigation frame  $X$ - $Y$ - $Z$  axes. The corresponding body frame for the WA frame in Fig. 3.6 is illustrated in Fig. 3.8. The rotations follow the same yaw  $\psi$ , pitch  $\theta$ , and roll  $\phi$  sequence as in Fig. 3.7.

### 3.1.6 Geographic Local Horizon Line of Sight

The line of sight (LOS) is defined in terms of azimuth  $Az$  and elevation  $El$  relative to a geographic frame (north–east–down or  $N$ – $E$ – $D$ ), as shown in Fig. 3.9. The first rotation is about  $D$  through the  $Az$  angle, then a rotation through  $El$  to establish the LOS axis.

## 3.2 Coordinate Frame Transformations

This section develops several coordinate frame transformation matrices defined in the preceding section. Because several ECEF frames are used, a subscript within parentheses is used to denote the axis about which the Earth's rotation is defined for the corresponding frames as a means to distinguish the associated matrix formed for one frame from that used for another.

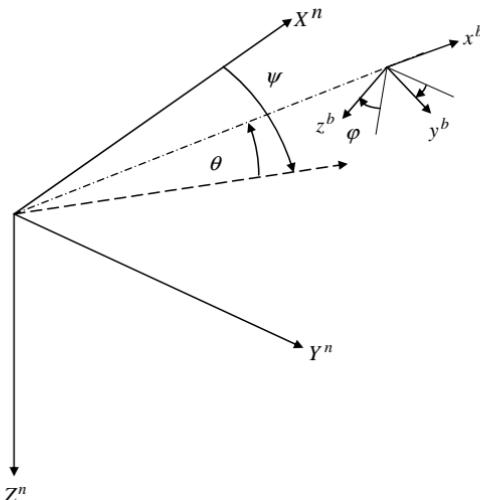
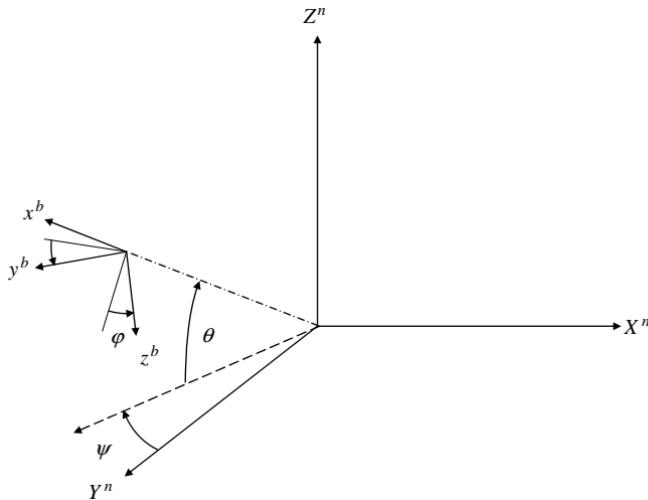
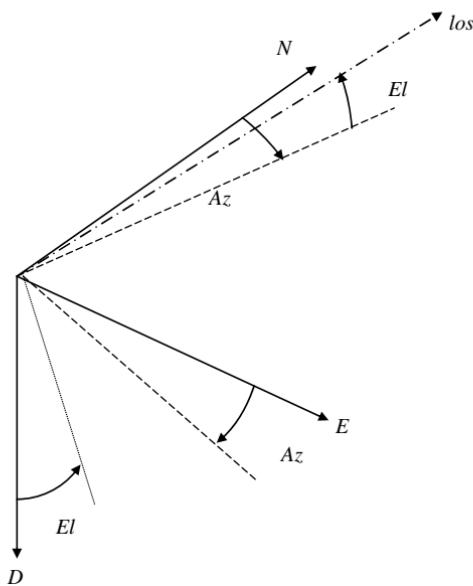


Fig. 3.7 Body coordinate frame for Fig. 3.4 WA frame.



**Fig. 3.8** Body coordinate frame for Fig. 3.6 WA frame.



**Fig. 3.9** Geographic horizon line-of-sight frame.

### 3.2.1 Earth-Centered to Earth-Fixed to Local Geodetic

Referring to Fig. 3.3, the transformation matrix from ECEF to local geodetic proceeds with the following sequence of two rotations:

*Rotation 1* (positive rotation  $\lambda$  about  $x_e$ ):

$$C_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\lambda & s\lambda \\ 0 & -s\lambda & c\lambda \end{bmatrix} \quad (3.3)$$

*Rotation 2* (positive rotation  $\phi$  about  $y'_e$ ):

$$C_2 = \begin{bmatrix} c\phi & 0 & -s\phi \\ 0 & 1 & 0 \\ s\phi & 0 & c\phi \end{bmatrix} \quad (3.4)$$

where  $y'_e$  represents the transformed  $y$  axis after rotation 1. The resulting ECEF to local geodetic (north–west–up) transformation matrix is

$$C_e^g(x) = \begin{bmatrix} c\phi & s\phi s\lambda & -s\phi c\lambda \\ 0 & c\lambda & s\lambda \\ s\phi & -c\phi s\lambda & c\phi c\lambda \end{bmatrix} \quad (3.5)$$

### 3.2.2 Earth-Centered, Earth-Fixed to Navigation Frame (Wander Azimuth)

This transformation matrix requires an additional transformation from the local geodetic to the navigation frame through the WA angle  $\alpha$ , as shown in Fig. 3.6.

$$\begin{aligned} C_e^n(x) &= C_g^n(x) C_e^g(x) \\ &= \begin{bmatrix} c\alpha & s\alpha & 0 \\ -s\alpha & c\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} C_e^g(x) \end{aligned} \quad (3.6)$$

Or,

$$C_e^n(x) = \begin{bmatrix} c\alpha c\phi & c\alpha s\phi s\lambda + s\alpha c\lambda & -c\alpha s\phi c\lambda + s\alpha s\lambda \\ -s\alpha c\phi & -s\alpha s\phi s\lambda + c\alpha c\lambda & s\alpha s\phi c\lambda + c\alpha s\lambda \\ s\phi & -c\phi s\lambda & c\phi c\lambda \end{bmatrix} \quad (3.7)$$

Note that

$$C_n^e(x) = [C_e^n(x)]^T \quad (3.8)$$

### 3.2.3 Earth-Centered, Earth-Fixed to Local Geodetic

Referring to Fig. 3.1, the transformation matrix from ECEF to local geodetic proceeds with the following sequence of three rotations:

*Rotation 1* [positive rotation (right-hand sense)  $\lambda$  about  $z_e$ ]:

$$C_1 = \begin{bmatrix} c\lambda & s\lambda & 0 \\ -s\lambda & c\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.9)$$

*Rotation 2* (positive rotation  $\phi$  about  $y'_e$ , but with a change in sign to change the sense):

$$C_2 = \begin{bmatrix} c\phi & 0 & s\phi \\ 0 & 1 & 0 \\ -s\phi & 0 & c\phi \end{bmatrix} \quad (3.10)$$

*Rotation 3* (align axes from up–east–north to north–east–down):

$$C_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad (3.11)$$

or

$$C_e^g(z) = \begin{bmatrix} -s\phi c\lambda & -s\phi s\lambda & c\phi \\ -s\lambda & c\lambda & 0 \\ -c\phi c\lambda & -c\phi s\lambda & -s\phi \end{bmatrix} \quad (3.12)$$

### 3.2.4 Earth-Centered Inertial to Local Geodetic Frame

The rotation from ECI to ECEF, as a result the Earth's rotation, is defined by the direction of the Earth's rotation. The angle resulting from this rotation is defined as

$$\theta = \omega_{i/e} t \quad (3.13)$$

The preceding transformation from ECEF to local geodetic is postmultiplied by an additional transformation resulting from a rotation about the  $z$  axis through the

angle  $\theta$  as

$$\begin{aligned}
 C_i^g(z) &= C_e^g(z)C_i^e(z) \\
 &= C_e^g(z) \begin{bmatrix} c\theta & s\theta & 0 \\ -s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= C_3C_2 \begin{bmatrix} c\lambda & s\lambda & 0 \\ -s\lambda & c\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta & s\theta & 0 \\ -s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= C_3C_2 \begin{bmatrix} c(\lambda + \theta) & s(\lambda + \theta) & 0 \\ -s(\lambda + \theta) & c(\lambda + \theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.14}
 \end{aligned}$$

or

$$C_i^g(z) = \begin{bmatrix} -s\phi c(\lambda + \theta) & -s\phi s(\lambda + \theta) & c\phi \\ -s(\lambda + \theta) & c(\lambda + \theta) & 0 \\ -c\phi c(\lambda + \theta) & -c\phi s(\lambda + \theta) & -s\phi \end{bmatrix} \tag{3.15}$$

### 3.2.5 Body-to-Navigation Frame

Referring to Fig. 3.7, the transformation from the  $X-Y-Z$  navigation frame to the body  $x-y-z$  axes proceeds through yaw, pitch, and roll rotations. This transformation was developed in Sec. 2.4. This sequence of rotations produces the following transposed transformation matrix:

$$C_b^n(z) = \begin{bmatrix} c\theta c\psi_{az} & -c\varphi s\psi_{az} + s\varphi s\theta c\psi_{az} & s\varphi s\psi_{az} + c\varphi s\theta c\psi_{az} \\ c\theta s\psi_{az} & c\varphi c\psi_{az} + s\varphi s\theta s\psi_{az} & -s\varphi c\psi_{az} + c\varphi s\theta s\psi_{az} \\ -s\theta & s\varphi c\theta & c\varphi c\theta \end{bmatrix} \tag{3.16}$$

For the navigation frame shown in Fig. 3.8, the body-to-navigation frame transformation matrix is developed using the reverse of the sequences just used. The transformation matrix is produced through a sequence of four rotations. The first three transform from the body to an intermediate navigation frame that is consistent with the body axis illustrated in Fig. 3.8. That is, with zero rotations, the  $x$ ,  $y$ , and  $z$  axes for the two frames are parallel. The resulting intermediate navigation frame is then rotated about its  $x$  axis by 180 deg, which is common to both intermediate and final navigation frame axes. The required transformation matrix proceeds with the following sequence of rotations. Note that although the rotations are indicated as negative, the individual rotations are formed as positive rotations, and then the sign of the angle is changed to change the sense of the rotation.

*Rotation 1* ( $-\varphi$  about  $x_b$ ):

$$C_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\varphi & -s\varphi \\ 0 & s\varphi & c\varphi \end{bmatrix} \quad (3.17)$$

*Rotation 2* ( $-\theta$  about  $y'_b$ ):

$$C_2 = \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \quad (3.18)$$

*Rotation 3* ( $-\psi_{az}$  about  $z''_b$ ):

$$C_3 = \begin{bmatrix} c\psi_{az} & -s\psi_{az} & 0 \\ s\psi_{az} & c\psi_{az} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.19)$$

*Rotation 4* (180 deg about  $x'''_b$ ):

$$C_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (3.20)$$

The body-to-navigation frame transformation matrix is

$$C_b^n(x) = \begin{bmatrix} c\psi_{az}c\theta & c\psi_{az}s\theta s\varphi - s\psi_{az}c\varphi & c\psi_{az}s\theta c\varphi + s\psi_{az}s\varphi \\ -s\psi_{az}c\theta & -s\psi_{az}s\theta s\varphi - c\psi_{az}c\varphi & c\psi_{az}s\varphi - s\psi_{az}s\theta c\varphi \\ s\theta & -c\theta s\varphi & -c\theta c\varphi \end{bmatrix} \quad (3.21)$$

### 3.2.6 Geodetic to Line of Sight

Referring to Fig. 3.9, the transformation matrix from geodetic to line of sight proceeds with the following sequence of two rotations.

*Rotation 1* ( $Az$  about  $D$ ):

$$C_1 = \begin{bmatrix} cAz & sAz & 0 \\ -sAz & cAz & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.22)$$

*Rotation 2* ( $El$  about  $E'$ ):

$$C_2 = \begin{bmatrix} cEl & 0 & -sEl \\ 0 & 1 & 0 \\ sEl & 0 & cEl \end{bmatrix} \quad (3.23)$$

The resulting geographic to line-of-sight transformation matrix is

$$C_g^{LOS} = \begin{bmatrix} cElcAz & cElsAz & -sEl \\ -sAz & cAz & 0 \\ sElcAz & sElsAz & cEl \end{bmatrix} \quad (3.24)$$

### Example 3.1 Line-of-Sight Angles and Linearization Using Relative Positions

Define the relative position of an object being optically tracked with respect to the tracking device's position as

$$\Delta x = x - x_s$$

$$\Delta y = y - y_s$$

$$\Delta z = z - z_s$$

where  $x$ ,  $y$ , and  $z$  are the object's local position coordinates and  $x_s$ ,  $y_s$ , and  $z_s$  are the tracking system's local position coordinates. It is assumed that the tracking device is mechanized with elevation as the inner gimbal. The equations for these measurements are obtained by a sequence of rotations, first azimuth and second elevation, transforming the Cartesian axis components into the line-of-sight frame as

$$\begin{bmatrix} \Delta x' \\ \Delta y'' \\ \Delta z'' \end{bmatrix} = \begin{bmatrix} cEl & 0 & -sEl \\ 0 & 1 & 0 \\ sEl & 0 & cEl \end{bmatrix} \begin{bmatrix} cAz & sAz & 0 \\ -sAz & cAz & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}$$

where  $cEl$ ,  $sEl$  are the cosine and sine of the elevation gimbal angle and  $cAz$ ,  $sAz$  are the cosine and sine of the azimuth gimbal angle. Completing the preceding matrix multiplications yields the following. In this equation, the component of position along the line of sight is range, and the others are defined as zero—defining the line-of-sight angles.

$$\begin{bmatrix} \rho \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} cElcAz & cElsAz & -sEl \\ -sAz & cAz & 0 \\ sElcAz & sElsAz & cEl \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}$$

where  $\rho$  is the range that is not provided by an optical tracking system. This matrix equation is expanded into component form for each of the last two rows. From these two equations, expressions for the azimuth and elevation can be obtained in terms of the relative positions.

Table 3.1 summarizes the azimuth and elevation equations and their partial derivatives with respect to the relative positions.

**Table 3.1** *Az* and *El* measurements and partial derivatives

Measurement equation	Partial derivatives with respect to ( )		
	x	y	z
$\tan Az = \frac{\Delta y}{\Delta x}$	$\frac{-\Delta y}{\Delta x^2 + \Delta y^2}$	$\frac{\Delta x}{\Delta x^2 + \Delta y^2}$	—
$\tan El = \frac{-\Delta z}{cAz\Delta x + sAz\Delta y}$	$\frac{cAz\Delta z}{( )^2 + \Delta z^2}$	$\frac{sAz\Delta z}{( )^2 + \Delta z^2}$	$\frac{a( )}{( )^2 + \Delta z^2}$

$$a( ) = cAz\Delta x + sAz\Delta y.$$

### Example 3.2 Wander Angle Rate for $C_e^n(x)$

The equation describing the rate of change in the Earth-to-navigation frame direction cosine matrix is established using the general form presented in Sec. 2.4. The elements of the  $C_e^n(x)$  matrix are given in Eqs. (3.7) and (3.8). This example develops the expression in Eq. (3.2) for the WA angle rate.

The rate of change of the direction cosine matrix is (momentarily dropping the preceding  $x$  subscript)

$$\dot{C}_e^n = -\Omega_{e/n}^n C_e^n$$

The navigation-to-Earth angular rotation vector, referenced in the navigation frame, is redefined for notational purposes as

$$\omega_{e/n}^n \equiv \begin{bmatrix} \rho_x \\ \rho_y \\ \rho_z \end{bmatrix}$$

Substituting the elements of this vector into the proper elements for the skew-symmetric form of the preceding vector cross product yields, for selected elements,

$$\dot{C}_{e\ 11}^n = \rho_z C_{e\ 21}^n - \rho_y C_{e\ 31}^n$$

$$\dot{C}_{e\ 31}^n = \rho_y C_{e\ 11}^n - \rho_x C_{e\ 21}^n$$

$$\dot{C}_{e\ 32}^n = \rho_y C_{e\ 12}^n - \rho_x C_{e\ 22}^n$$

Referring to the elements of the direction cosine matrix for the left and right side of these equations, these equations can be rewritten as

$$-sac\phi\dot{\alpha} - cas\phi\dot{\phi} = -sac\phi\rho_z - s\phi\rho_y$$

$$c\phi\dot{\phi} = cac\phi\rho_y + sac\phi\rho_x$$

$$s\phi s\lambda\dot{\phi} - c\phi c\lambda\dot{\lambda} = (cas\phi s\lambda + sac\lambda)\rho_y - (-sas\phi s\lambda + cac\lambda)\rho_x$$

Solving the second equation for  $\dot{\phi}$  yields

$$\dot{\phi} = c\alpha\rho_y + s\alpha\rho_x$$

Substituting this equation into the third equation yields

$$\dot{\lambda} = \frac{-(s\alpha\rho_y - c\alpha\rho_x)}{c\phi}$$

Substituting the equation for  $\dot{\phi}$  into the first equation yields

$$(\dot{\alpha} - \rho_z)c\phi = -(-s\alpha\rho_y + c\alpha\rho_x)s\phi$$

which, after using the equation for  $\dot{\lambda}$ , gives the equation for WA rate

$$\dot{\alpha} = -\dot{\lambda} \sin \phi + \rho_z$$

To obtain Eq. (3.2) requires that the angular rate about the  $z$  axis be zero:

$$\rho_z = 0$$

### Example 3.3 Angular Position and Latitude/Longitude/Wander Angle Errors

This example introduces the concept of an external error vector, angular position error, associated with the position direction cosine matrix. The angular position error vector  $\delta\theta$  is related to the computed or corrupted DCM,  $\bar{C}_e^n(x)$  (indicated with an overbar) and true DCM  $C_e^n(x)$  (without the overbar) as in the following equation:

$$\bar{C}_e^n \equiv [I - (\delta\theta \times)] C_e^n$$

where

$$[I - (\delta\theta \times)] = \begin{bmatrix} 1 & \delta\theta_z & -\delta\theta_y \\ -\delta\theta_z & 1 & \delta\theta_x \\ \delta\theta_y & -\delta\theta_x & 1 \end{bmatrix}$$

The  $C_e^n(x)$  matrix from Sec. 3.2 is

$$C_e^n(x) = \begin{bmatrix} c\alpha c\phi & c\alpha s\phi s\lambda + s\alpha c\lambda & -c\alpha s\phi c\lambda + s\alpha s\lambda \\ -s\alpha c\phi & -s\alpha s\phi s\lambda + c\alpha c\lambda & s\alpha s\phi c\lambda + c\alpha s\lambda \\ s\phi & -c\phi s\lambda & c\phi c\lambda \end{bmatrix}$$

This form holds whether the matrix is computed or true. If computed, then the corresponding latitude, longitude, and wander angles can be expressed in terms

of their true value plus error as

$$\begin{aligned}\bar{\phi} &= \phi + \delta\phi \\ \bar{\lambda} &= \lambda + \delta\lambda \\ \bar{\alpha} &= \alpha + \delta\alpha\end{aligned}$$

The objective is to relate the error angles in these equations to the elements of angular position error vector  $\delta\theta$ . It is assumed that  $\delta\theta$  is given and expressions for  $\delta\phi$ ,  $\delta\lambda$ , and  $\delta\alpha$  are to be obtained. This is accomplished by selecting elements in the computed  $C_e^n(x)$  matrix, expanding them in terms of the trigonometric relationships for sums of angles (assuming small error angles), and equating those expressions to their corresponding terms from the matrix multiplication in the first equation just given.

Beginning with the  $C_{e\ zx}^n$  term,

$$s\bar{\phi} = s\phi + c\phi\delta\phi = C_{e\ zx}^n + \delta\theta_y C_{e\ xx}^n - \delta\theta_x C_{e\ yx}^n$$

This implies, from the definition of  $C_e^n(x)$  just given,

$$\begin{aligned}c\phi\delta\phi &= \delta\theta_y C_{e\ xx}^n - \delta\theta_x C_{e\ yx}^n \\ &= \delta\theta_y(cac\phi) - \delta\theta_x(-sac\phi)\end{aligned}$$

or, after canceling the  $c\phi$  term, yields the following equation for latitude error:

$$\delta\phi = \delta\theta_y c\alpha + \delta\theta_x s\alpha$$

Next, the  $C_{e\ zy}^n$  elements are used to solve to longitude error

$$-c\bar{\phi}s\bar{\lambda} = -c\phi s\lambda + s\phi s\lambda\delta\phi - c\phi c\lambda\delta\lambda = C_{e\ zy}^n + \delta\theta_y C_{e\ xy}^n - \delta\theta_x C_{e\ yy}^n$$

which becomes

$$s\phi s\lambda\delta\phi - c\phi c\lambda\delta\lambda = \delta\theta_y C_{e\ xy}^n - \delta\theta_x C_{e\ yy}^n$$

From the definition of  $C_e^n(x)$  and the equation for  $\delta\phi$  just given

$$\begin{aligned}s\phi s\lambda(\delta\theta_y c\alpha + \delta\theta_x s\alpha) - c\phi c\lambda\delta\lambda &= \delta\theta_y(cas\phi s\lambda + sac\lambda) \\ &\quad - \delta\theta_x(-sas\phi s\lambda + cac\lambda)\end{aligned}$$

or, after expanding terms,

$$\begin{aligned}cas\phi s\lambda\delta\theta_y + sas\phi s\lambda\delta\theta_x - c\phi c\lambda\delta\lambda &= cas\phi s\lambda\delta\theta_y + sac\lambda\delta\theta_y \\ &\quad + sas\phi s\lambda\delta\theta_x - cac\lambda\delta\theta_x\end{aligned}$$

Cancelling terms yields

$$-c\phi c\lambda \delta\lambda = s\alpha c\lambda \delta\theta_y - c\alpha c\lambda \delta\theta_x$$

After cancelling  $c\lambda$ , the following equation for longitude error  $\delta\lambda$  is obtained:

$$\delta\lambda = \frac{(c\alpha \delta\theta_x - s\alpha \delta\theta_y)}{c\phi}$$

The position errors can be related to the latitude and longitude errors. Combining equations for the latitude and longitude errors, the following equations for the angular position errors are obtained:

$$\begin{aligned}\delta\theta_x &= c\alpha c\phi \delta\lambda + s\alpha \delta\phi \\ \delta\theta_y &= -s\alpha c\phi \delta\lambda + c\alpha \delta\phi\end{aligned}$$

Finally, from the  $C_{e\ xx}^n$  term,

$$c\bar{\alpha} c\bar{\phi} = c\alpha c\iota - s\alpha c\phi \delta\alpha - c\alpha s\phi \delta\phi = C_{e\ xx}^n + \delta\theta_z C_{e\ yx}^n - \delta\theta_y C_{e\ zx}^n$$

which becomes, after cancelling terms,

$$-s\alpha c\phi \delta\alpha - c\alpha s\phi \delta\phi = \delta\theta_z C_{e\ yx}^n - \delta\theta_y C_{e\ zx}^n$$

From the equation for the angular position error  $\delta\theta_y$  just given,

$$\begin{aligned}-s\alpha c\phi \delta\alpha - c\alpha s\phi \delta\phi &= \delta\theta_z (-s\alpha c\phi) - (-s\alpha c\phi \delta\lambda + c\alpha \delta\phi)(s\phi) \\ &= -s\alpha c\phi \delta\theta_z + s\alpha c\phi s\phi \delta\lambda - c\alpha s\phi \delta\phi\end{aligned}$$

After cancelling terms, the following equation for WA error  $\delta\alpha$  is obtained:

$$\delta\alpha = \delta\theta_z - s\phi \delta\lambda$$

This example illustrates the relationship between the external error associated with the position direction cosine matrix and errors in latitude, longitude, and wander-azimuth angles that are used to form this DCM.

### 3.3 Chapter Summary

Several coordinate systems that are currently used for integrated navigation systems were presented, and their associated transformation matrices were developed. The awareness of different coordinate systems and their usage should reinforce the notion that assumed coordinate system definitions need to be verified prior to using data defined in different systems and consistency in their use to ensure a successful navigation system.

The concept of position and attitude errors associated with their corresponding direction cosine matrices was presented. Linearized navigation state error equations for position and attitude errors, to be developed in Sec. 5.3, will be based on these error definitions.

### Problems

- 3.1** This exercise develops the relationships between  $C_e^n(y)$  matrix and latitude, longitude, and wander angles. Referring to Fig. 3.2, show that the corresponding navigation-to-Earth frame direction cosine matrix (transposed) is

$$C_e^n(y) = \begin{bmatrix} \cos\lambda - \sin\phi\sin\lambda & \sin\phi & -\cos\lambda - \sin\phi\cos\lambda \\ -\sin\lambda - \cos\phi\sin\lambda & \cos\phi & \sin\lambda - \cos\phi\cos\lambda \\ \cos\lambda & \sin\phi & \cos\phi\cos\lambda \end{bmatrix}$$

where

$$C_n^e(y) = [C_e^n(y)]^T$$

and the following relationships:

$$\omega_{i/e}^n = C_e^n(y) \begin{bmatrix} 0 \\ \omega_{i/e} \\ 0 \end{bmatrix} = \omega_{i/e} \begin{bmatrix} C_e^n_{21} \\ C_e^n_{22} \\ C_e^n_{23} \end{bmatrix}$$

$$\phi = \sin^{-1}(C_n^{e23})$$

$$\lambda = \tan^{-1}\left(\frac{C_n^{e13}}{C_n^{e33}}\right)$$

$$\alpha = \tan^{-1}\left(\frac{C_n^{e21}}{C_n^{e22}}\right)$$

- 3.2** This exercise develops the wander angle rate for  $C_n^e(y)$ . Show that the following expressions hold:

$$\dot{\phi} = s\alpha\rho_y - c\alpha\rho_x$$

$$\dot{\lambda} = \frac{(c\alpha\rho_y + s\alpha\rho_x)}{c\phi}$$

$$\dot{\alpha} = -\dot{\lambda} \sin\phi + \rho_z$$

Note that the expression for the WA rate here is the same as that obtained in Example 3.2.

- 3.3** This exercise examines the relationship between angular position and latitude/longitude/wander angle errors. Show that the following expressions hold for the ECEF coordinate frame  $C_n^e(y)$ :

$$\delta\phi = \delta\theta_y s\alpha - \delta\theta_x c\alpha$$

$$\delta\lambda = \frac{(s\alpha\delta\theta_x + c\alpha\delta\theta_y)}{c\phi}$$

$$\delta\theta_x = s\alpha c\phi \delta\lambda - c\alpha \delta\phi$$

$$\delta\theta_y = c\alpha c\phi \delta\lambda + s\alpha \delta\phi$$

$$\delta\alpha = \delta\theta_z - s\phi \delta\lambda$$

Note that the expression for the WA error is again the same as that obtained in Example 3.3.

- 3.4** This exercise continues to relate the external error vector, tilt errors, to errors associated with the Euler angles associated with the attitude direction cosine matrix  $C_b^n(x)$ . That is, tilt error vector  $\boldsymbol{\phi}$  is to be related to yaw, pitch, and roll errors— $\delta\psi$ ,  $\delta\theta$ , and  $\delta\varphi$ . Tilt errors are defined in terms of the computed and true body-to-navigation frame direction cosine matrix

$$\bar{C}_b^n \equiv [I - (\boldsymbol{\phi} \times)] C_b^n$$

where

$$[I - (\boldsymbol{\phi} \times)] = \begin{bmatrix} 1 & \phi_z & -\phi_y \\ -\phi_z & 1 & \phi_x \\ \phi_y & -\phi_x & 1 \end{bmatrix}$$

The  $C_b^n(x)$  matrix is given in Eq. (3.21) as

$$C_b^n(x) = \begin{bmatrix} c\psi_{az}c\theta & c\psi_{az}s\theta s\varphi - s\psi_{az}c\varphi & c\psi_{az}s\theta c\varphi + s\psi_{az}s\varphi \\ -s\psi_{az}c\theta & -s\psi_{az}s\theta s\varphi - c\psi_{az}c\varphi & c\psi_{az}s\gamma - s\psi_{az}s\theta c\varphi \\ s\theta & -c\theta s\varphi & -c\theta c\varphi \end{bmatrix}$$

This equation is assumed to hold whether it is computed or true. If computed, then the corresponding yaw, pitch, and roll angles can be expressed in terms of their true values plus error

$$\bar{\psi}_{az} = \psi_{az} + \delta\psi$$

$$\bar{\theta} = \theta + \delta\theta$$

$$\bar{\varphi} = \varphi + \delta\varphi$$

The objective is to relate the preceding error angles to the tilt errors. This is accomplished by selecting elements in the computed  $C_b^n(x)$  matrix, expanding them in terms of the trigonometric relationships for sums of angles, and equating those expressions to their corresponding terms from the matrix multiplication given in the first equation just given.

Beginning with the  $C_{bx}^n$  terms,

$$s\bar{\theta} = s\theta + c\theta\delta\theta = C_{bx}^n + \phi_y C_{xx}^n - \phi_x C_{yx}^n$$

This implies from the definition of  $C_b^n(x)$

$$\begin{aligned} c\theta\delta\theta &= \phi_y C_{xx}^n - \phi_x C_{yx}^n \\ &= \phi_y(c\psi_{az}c\theta) - \phi_x(-s\psi_{az}c\theta) \end{aligned}$$

or, after cancelling the  $c\theta$  terms,

$$\delta\theta = c\psi_{az}\phi_y + s\psi_{az}\phi_x$$

Continue in a similar fashion to obtain the following:

$$\delta\psi_{az} = \phi_z - (c\psi_{az}\phi_x - s\psi_{az}\phi_y)t\theta$$

and

$$\delta\varphi = \frac{s\psi_{az}\phi_y - c\psi_{az}\phi_x}{c\theta}$$

- 3.5** Referring to Fig. 3.1, show that the corresponding navigation-to-Earth frame direction cosine matrix is

$$C_e^n(z) = \begin{bmatrix} -cas\phi c\lambda - sas\lambda & sas\phi c\lambda - cas\lambda & -c\phi c\lambda \\ -cas\phi s\lambda + sac\lambda & sas\phi s\lambda + cac\lambda & -c\phi s\lambda \\ cac\phi & -sac\phi & -s\phi \end{bmatrix}$$

and the following relationship:

$$\boldsymbol{\omega}_{i/e}^n \equiv \boldsymbol{\Omega} = C_e^n(z) \begin{bmatrix} 0 \\ 0 \\ \omega_{i/e} \end{bmatrix}$$

- 3.6** By paralleling the steps in Example 3.2, obtain the following expression for the WA rate for the ECEF coordinate frame  $C_e^n(z)$ :

$$\dot{\alpha} = \dot{\lambda} \sin \phi + \rho_z$$

- 3.7** In this exercise, body-referenced tilt errors are related to Euler-angle errors for the  $C_b^n(z)$  matrix. Define body-referenced tilt errors  $\boldsymbol{\phi}^b$  as the result of the following product:

$$\begin{aligned}(C_b^n)^T \bar{C}_b^n &= I - C_n^b(\boldsymbol{\phi} \times) C_b^n \\ &= I - (\boldsymbol{\phi}^b \times)\end{aligned}$$

Establish the following relationships for the components of  $\boldsymbol{\phi}^b$ :

$$\begin{aligned}\phi_x^b &= -\delta\varphi - s\theta\delta\psi_{az} \\ \phi_y^b &= -c\varphi\delta\theta - s\varphi c\theta\delta\psi_{az} \\ \phi_z^b &= s\varphi\delta\theta - c\varphi c\theta\delta\psi_{az}\end{aligned}$$

- 3.8** This exercise develops relationships between the attitude error and quaternion error from the preceding chapter. From Problem 2.10, obtain the following equations for quaternion errors:

$$\begin{aligned}\delta C_{11} + \delta C_{22} + \delta C_{33} &= 8q_0\delta q_0 \\ \delta q_1 &= \frac{\delta C_{32} - \delta C_{23}}{4q_0} - \frac{C_{32} - C_{23}}{4q_0^2} \delta q_0 \\ &= \frac{\delta C_{32} - \delta C_{23}}{4q_0} - \frac{q_1}{q_0} \delta q_0 \\ \delta q_2 &= \frac{\delta C_{13} - \delta C_{31}}{4q_0} - \frac{q_2}{q_0} \delta q_0 \\ \delta q_3 &= \frac{\delta C_{21} - \delta C_{12}}{4q_0} - \frac{q_3}{q_0} \delta q_0\end{aligned}$$

Define a computed DCM similarly as just shown

$$\begin{aligned}\bar{C}_b^i &\equiv [I - (\boldsymbol{\phi} \times)] C_b^i \\ &= C_b^i - (\boldsymbol{\phi} \times) C_b^i\end{aligned}$$

and obtain the error matrix

$$\delta C_b^i = -(\boldsymbol{\phi} \times) C_b^i$$

Expand terms in this error matrix, and insert them into the quaternion error equations just given to obtain the following relationship between quaternion errors and attitude errors:

$$\begin{bmatrix} \delta q_1 \\ \delta q_2 \\ \delta q_3 \\ \delta q_0 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -q_0 & -q_3 & q_2 \\ q_3 & -q_0 & -q_1 \\ -q_2 & q_1 & -q_0 \\ q_1 & q_2 & q_3 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}$$

which can be rewritten, using the results from Exercise 2.11, associating the  $y$  frame with the moving body frame and the  $x$  frame as the fixed inertial frame,

$$\delta \mathbf{q} = \frac{1}{2} Z^b \boldsymbol{\phi}$$

where

$$Z^b \equiv \begin{bmatrix} -q_0 & -q_3 & q_2 \\ q_3 & -q_0 & -q_1 \\ -q_2 & q_1 & -q_0 \\ q_1 & q_2 & q_3 \end{bmatrix}$$



## 4 Earth Models

This chapter presents two important modeling elements of an integrated navigation system: ellipsoid geometry and ellipsoid gravity. Like the different navigation frames presented in Chapter 3, there are many models for the Earth's geometry and gravity. These models are based on parameters that have been assigned different values for different uses. For integrated navigation systems, the Earth's shape is modeled by simple oblate spheroid, and the gravity models reflect a similar simplification.

The local-level frames discussed in Chapter 3 are to be maintained as locally level as a vehicle moves over the Earth's surface. This movement results in an angular rotation defined by the Earth's geometrical shape, that is, its radius of curvature. The numerical integration of this motion yields the vehicle position. An accurate model of the Earth's shape is necessary so that an accurate position results from that integration. The radius of curvature expressions developed in this chapter are used in Chapter 5's developments of navigation system dynamic equations.

The Earth's gravity influences accelerations sensed by the inertial system's instruments. Models for gravity are maintained within the navigation system to determine what part of the sensed acceleration is caused by vehicle dynamics and what is caused by the Earth's gravitational attraction. The most widely used current Earth model is the World Geodetic System (WGS-84) [4].

Approximations to the shape and gravity models that are later used in the linearization of navigation state equations presented in Chapter 5 are developed.

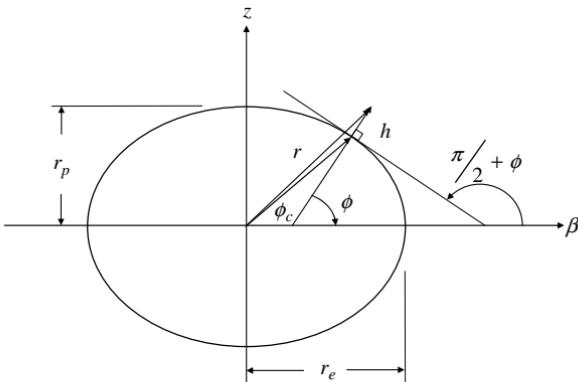
### 4.1 Ellipsoid Geometry

Earth-shape model parameters are illustrated in Fig. 4.1, which shows the elliptical cross section for the ellipsoid. The  $z$  axis is along the rotation axis, and  $\beta$  is a radial distance in the  $x$ - $y$  plane of the equator.

Several geometrical parameters used in definitions for Earth-shape models are illustrated in Fig. 4.1. The following relationships for eccentricity  $\varepsilon$ , ellipticity  $e$ , and flattening  $f$  define the ellipsoid's shape and other variables:

$$\varepsilon = \left(1 - \frac{r_p^2}{r_e^2}\right)^{1/2} \quad (4.1)$$

$$e = 1 - \frac{r_p}{r_e} \quad (4.2)$$



**Fig. 4.1 Earth-shape model geometry.**

$$f = \frac{1}{e} \quad (4.3)$$

#### 4.1.1 Earth-Centered X-Y-Z Coordinates from Latitude and Longitude

Variables of interest include  $X$ - $Y$ - $Z$  ECEF position coordinates based on surface-referenced latitude, longitude and altitude coordinates. These coordinates as affected by the Earth-shape model parameters, will be developed in the following.

The Earth's surface, as shown in Fig. 4.1, is described by the following equation for an ellipse:

$$\frac{\beta^2}{r_e^2} + \frac{z^2}{r_p^2} = 1 \quad (4.4)$$

To determine  $\beta$  and  $z$  in terms of the preceding shape parameters and the geodetic latitude  $\phi$ , first the geocentric latitude  $\phi_c$  is determined. The local slope, as shown in Fig. 4.1, is found from the differential of Eq. (4.4).

$$\frac{2\beta d\beta}{r_e^2} + \frac{2z dz}{r_p^2} = 0 \quad (4.5)$$

or

$$\frac{dz}{d\beta} = -\frac{\beta r_p^2}{z r_e^2} \quad (4.6)$$

From inspection of Fig. 4.1,

$$\frac{dz}{d\beta} = \tan\left(\frac{\pi}{2} + \phi\right) = -\frac{1}{\tan \phi} \quad (4.7)$$

or, combining the results from the last two equations,

$$-\frac{1}{\tan \phi} = -\frac{\beta r_p^2}{z r_e^2} \quad (4.8)$$

From the definition in Eq. (4.1),

$$\frac{r_p^2}{r_e^2} = -(e^2 - 1) \quad (4.9)$$

Then,

$$\frac{z}{\beta} = (1 - e^2) \tan \phi \quad (4.10)$$

Again, by inspection of Fig. 4.1,

$$\frac{z}{\beta} = \tan \phi_c \quad (4.11)$$

Equating Eqs. (4.10) and (4.11), the following equation for the geocentric latitude is obtained in terms of the eccentricity and geodetic latitude:

$$\tan \phi_c = (1 - e^2) \tan \phi \quad (4.12)$$

From Eq. (4.4),

$$\beta^2 = r_e^2 \left(1 - \frac{z^2}{r_p^2}\right) = r_e^2 - \frac{r_e^2}{r_p^2} z^2 = r_e^2 - \frac{1}{(1 - e^2)} z^2 \quad (4.13)$$

Combining this equation with Eq. (4.10)

$$\frac{z}{\beta} = (1 - e^2) \tan \varphi \quad \Rightarrow \quad z = \beta(1 - e^2) \tan \varphi \quad \Rightarrow \quad z^2 = \beta^2(1 - e^2)^2 \tan^2 \varphi$$

yields for  $\beta^2$

$$\begin{aligned}\beta^2 &= r_e^2 - \frac{1}{(1 - \varepsilon^2)} \beta^2 (1 - \varepsilon^2)^2 \tan^2 \varphi \\ &= r_e^2 - \beta^2 (1 - \varepsilon^2) \tan^2 \varphi \\ &= \frac{r_e^2}{[1 + (1 - \varepsilon^2) \tan^2 \varphi]}\end{aligned}$$

and solving for beta

$$\begin{aligned}\beta &= \frac{r_e}{[1 + (1 - \varepsilon^2) \tan^2 \varphi]^{1/2}} \\ &= \frac{r_e}{[1 + \tan^2 \varphi - \varepsilon^2 \tan^2 \varphi]^{1/2}} \\ &= \frac{r_e}{[1 + \sin^2 \varphi / \cos^2 \varphi - \varepsilon^2 (\sin^2 \varphi / \cos^2 \varphi)]^{1/2}} \\ &= \frac{r_e}{[(\sin^2 \varphi + \cos^2 \varphi) / \cos^2 \varphi - \varepsilon^2 (\sin^2 \varphi / \cos^2 \varphi)]^{1/2}}\end{aligned}$$

Or

$$\beta = \frac{r_e \cos \phi}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} \quad (4.14)$$

And  $z$  can be obtained by substituting Eq. (4.14) into Eq. (4.10):

$$\begin{aligned}z &= \beta(1 - \varepsilon^2) \tan \varphi \\ &= \frac{r_e \cos \varphi}{[1 - \varepsilon^2 \sin^2 \varphi]^{1/2}} (1 - \varepsilon^2) \frac{\sin \varphi}{\cos \varphi}\end{aligned}$$

Or

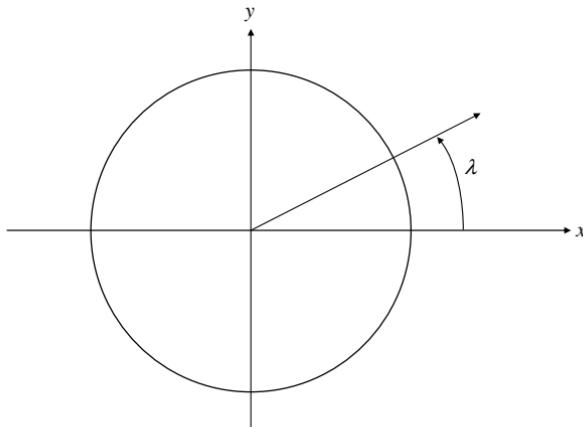
$$z = \frac{r_e (1 - \varepsilon^2) \sin \phi}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} \quad (4.15)$$

From these equations, the  $X-Y-Z$  position coordinates in the ECEF frame of a point of the surface can be determined.

From Fig. 4.2, the position in the  $x-y$  plane is given in terms of the radial position  $\beta$  and longitude  $\lambda$  as

$$x = \beta \cos \lambda \quad (4.16)$$

$$y = \beta \sin \lambda \quad (4.17)$$



**Fig. 4.2** Earth-shape model  $X-Y$  (equatorial) plane.

From these equations and the preceding results, the following are obtained:

$$x = \frac{r_e \cos \phi \cos \lambda}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} \quad (4.18)$$

$$y = \frac{r_e \cos \phi \sin \lambda}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} \quad (4.19)$$

Approximating the altitude above the surface following Britting [1], equations for the  $X-Y-Z$  coordinates become

$$x^{ECEF} = \left[ \frac{r_e}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} + h \right] \cos \phi \cos \lambda \quad (4.20)$$

$$y^{ECEF} = \left[ \frac{r_e}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} + h \right] \cos \phi \sin \lambda \quad (4.21)$$

$$z^{ECEF} = \left[ \frac{r_e(1 - \varepsilon^2)}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} + h \right] \sin \phi \quad (4.22)$$

#### 4.1.2 Radii of Curvature

The radii of curvature along lines of constant longitude and latitude are defined as  $R_{\text{meridian}}$  and  $R_{\text{normal}}$ , respectively. Terrestrial navigation systems are mechanized or implemented such that the local-level frame is maintained while the vehicle is changing position. These radii of curvature will be used later to determine the change in the navigation frame's orientation with change in position.

From calculus, the radius of curvature of an arc, in this case of constant longitude, is given by

$$R_{\text{meridian}} = \left[ 1 + \left( \frac{dz}{d\beta} \right)^2 \right]^{3/2} \Bigg/ \pm \frac{d^2 z}{d\beta^2} \quad (4.23)$$

Evaluating the second derivative from Eq. (4.6),

$$\frac{d^2 z}{d\beta^2} = -\frac{r_p^4}{r_e^2 z^3} \quad (4.24)$$

Substituting this result with Eqs. (4.6) and (4.9) into Eq. (4.23) yields

$$R_{\text{meridian}} = \frac{[z^2 + (1 - \varepsilon^2)^2 \beta^2]^{1/2}}{r_p^2 (1 - \varepsilon^2)} \quad (4.25)$$

After substituting the expressions for  $\beta$  in Eq. (4.14) and  $z$  in Eq. (4.15), this radius of curvature becomes

$$R_{\text{meridian}} = \frac{r_e (1 - \varepsilon^2)}{(1 - \varepsilon^2 \sin^2 \phi)^{3/2}} \quad (4.26)$$

The radius at the ellipsoid surface along constant latitudes is  $\beta$ . The radius of curvature normal to the ellipsoid surface at the point of tangency, at a given latitude,

**Table 4.1 Reference ellipsoid constants [4]**

Reference ellipsoid	$r_e$ , m	$f$
Clarke 1866	6,378,206.4	294.9786982
Clarke 1880	6,378,249.145	294.465
International	6,378,388	297
Bessel	6,377,397.155	299.1528128
Everest	6,377,276.345	300.8017
Modified Everest	6,377,304.063	300.8017
Australian National	6,378,160	298.25
South American 1969	6,378,160	298.25
Airy	6,377,564.396	299.3249646
Modified Airy	6,377,340.189	299.3249646
Hough	6,378,270	297
Fischer 1960 (South Asia)	6,378,155	298.3
Fischer 1960 (Mercury)	6,378,166	298.3
Fischer 1968	6,378,150	298.3
WGS-60	6,378,165	298.3
WGS-66	6,378,145	298.25
WGS-72	6,378,135	298.26
WGS-84	6,378,137	298.257223563

**Table 4.2 WGS-84 ellipsoid constants [4]**

<i>Defining parameters</i>	<i>Value</i>
Equatorial radius $r_e$ , m	6,378,137
Angular velocity $\omega_{i/e}$ , rad/s	$7.292115 \times 10^{-5}$
Earth's gravitational constant $\mu$ , $\text{m}^3/\text{s}^2$	$3.986005 \times 10^{+14}$
Second gravitational constant $J_2$	$1.08263 \times 10^{-3}$
<i>Derived constants</i>	<i>Value</i>
Flattening $f$	298.257223563
Polar radius $r_p$ , m	6,356,752.3142
First eccentricity $\varepsilon$	0.0818191908426
Gravity at equator $g_{\text{WGS}_0}$ , $\text{m/s}^2$	9.7803267714
Gravity formula constant $g_{\text{WGS}_1}$	0.00193185138639
Mean value (normal) gravity $g$ , $\text{m/s}^2$	9.7976446561

is given as

$$R_{\text{normal}} = \frac{\beta}{\cos \phi} \quad (4.27)$$

From Eq. (4.14), this radius of curvature becomes

$$R_{\text{normal}} = \frac{r_e}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} \quad (4.28)$$

Constants for several reference ellipsoid models are presented in Table 4.1. The number of different models reflects the changing knowledge and different applications. Table 4.2 lists several defining constants and resulting derived constants, specifically for the WGS-84 model.

### Example 4.1 Latitude, Longitude, and Geographic Frame Inertial Rates

*Latitude and longitude rates.* Motion over the surface of the ellipsoid is along the arc defining the surface. The rate of change of latitude is along the meridian and is governed by the curvature along this arc as

$$\dot{\phi} = \frac{v_{\text{north}}}{R_{\text{meridian}} + h}$$

The rate of change of longitude is found from

$$\dot{\lambda} = \frac{v_{\text{east}}}{\beta}$$

or, from Eq. (4.27),

$$\dot{\lambda} = \frac{v_{\text{east}}}{(R_{\text{normal}} + h) \cos \phi}$$

Note that this equation for longitude rate contains a mathematical singularity at the Earth's poles. If polar operations are to be considered in the navigation system's design, then other methods to evolve position must be chosen, for example, direction cosine matrices or quaternions. This form is used only in this example to illustrate the formation of rotation rate vectors.

**Local frame rate.** The longitude rate is an angular rate relative to the ECEF frame about the  $z$  axis shown in Fig. 3.1. A local-level  $g$  frame is formed with a nonzero latitude. The longitude rate is transformed into an intermediate frame  $g'$  transformed through the latitude  $\phi$ . This places the longitude rate rotation vector in the same coordinate frame as the latitude rate. These two rate vectors are added using the angular velocity addition theorem to obtain the angular rate vector in the intermediate  $g'$  frame as

$$\boldsymbol{\omega}_{e/g'}^{g'} = \begin{bmatrix} c\phi & 0 & s\phi \\ 0 & 1 & 0 \\ -s\phi & 0 & c\phi \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\lambda} \end{bmatrix} + \begin{bmatrix} 0 \\ -\dot{\phi} \\ 0 \end{bmatrix}$$

The intermediate  $g'$  frame is parallel to the geographic  $g$  frame; however, the orientation of the  $g'$  frame is not the same as that of the  $g$  frame. The rotation vector in the  $g'$  frame can be transformed into the  $g$  frame as follows:

$$\boldsymbol{\omega}_{e/g}^g = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \boldsymbol{\omega}_{e/g'}^{g'}$$

Substituting the preceding equations for  $\dot{\phi}$  and  $\dot{\lambda}$  into the equation for  $\boldsymbol{\omega}_{e/g'}^{g'}$  and using this change in orientation transformation, the Earth geographic frame angular rotation vector becomes

$$\boldsymbol{\omega}_{e/g}^g = \begin{bmatrix} \frac{v_{\text{east}}}{(R_n + h)} \\ \frac{-v_{\text{north}}}{(R_m + h)} \\ \frac{-v_{\text{east}}}{(R_n + h)} \tan \phi \end{bmatrix}$$

**Geographic frame inertial rate.** The Earth's rotation expressed in ECEF frame is

$$\boldsymbol{\omega}_{i/e}^e = \begin{bmatrix} 0 \\ 0 \\ \omega_{i/e} \end{bmatrix}$$

Transforming into the geographic frame as before, this Earth rate angular rotation vector becomes

$$\boldsymbol{\omega}_{i/e}^g = \begin{bmatrix} \omega_{i/e} \cos \phi \\ 0 \\ -\omega_{i/e} \sin \phi \end{bmatrix}$$

Applying the angular velocity addition theorem, the total angular velocity of the local-level geographic frame with respect to the inertial frame can be expressed as

$$\boldsymbol{\omega}_{i/g}^g = \boldsymbol{\omega}_{e/g}^g + \boldsymbol{\omega}_{i/e}^g$$

or

$$\boldsymbol{\omega}_{i/g}^g = \begin{bmatrix} \omega_{i/e} \cos \phi + \frac{v_{\text{east}}}{(R_n + h)} \\ \frac{-v_{\text{north}}}{(R_m + h)} \\ -\omega_{i/e} \sin \phi - \frac{v_{\text{east}}}{(R_n + h)} \tan \phi \end{bmatrix}$$

## 4.2 Ellipsoid Gravity

### 4.2.1 Gravitation Potential

The gravitation potential for the Earth is formulated in the following from [1]. Consider the geometry defined in Fig. 4.3. The differential potential is defined as being proportional to the elemental mass  $dm(\rho, \beta, \lambda)$  divided by the magnitude of the distance between this mass element and another point as

$$dU = G \frac{dm}{|\mathbf{r} - \boldsymbol{\rho}|} \quad (4.29)$$

In this figure, the mass element  $dm(\rho, \beta, \lambda)$ , located at  $\boldsymbol{\rho}$ , and the other point  $p(r, \phi, \theta)$ , located at  $\mathbf{r}$ , and their position dependencies are defined in spherical coordinates.

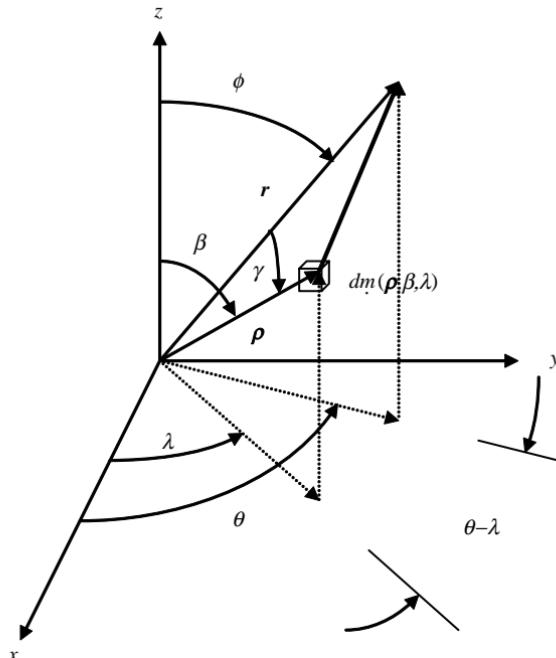


Fig. 4.3 Gravitation elements spherical geometry.

An expression for the magnitude of the distance in Eq. (4.29) is formed next. First, from the definition of vector dot products,

$$\mathbf{r} \cdot \boldsymbol{\rho} = |\mathbf{r}| |\boldsymbol{\rho}| \cos \gamma \quad (4.30)$$

then, the cosine of the angle  $\gamma$  becomes

$$\cos \gamma = \frac{\mathbf{r} \cdot \boldsymbol{\rho}}{|\mathbf{r}| |\boldsymbol{\rho}|} \quad (4.31)$$

The two positions indicated in the figure are expressed as

$$\mathbf{r} = r \begin{bmatrix} \sin \phi \cos \theta \\ \sin \phi \sin \theta \\ \cos \phi \end{bmatrix} \quad (4.32)$$

and

$$\boldsymbol{\rho} = \rho \begin{bmatrix} \sin \beta \cos \lambda \\ \sin \beta \sin \lambda \\ \cos \beta \end{bmatrix} \quad (4.33)$$

Using these forms, the vector dot product in Eq. (4.30) becomes

$$\begin{aligned} \mathbf{r} \cdot \boldsymbol{\rho} &= r\rho(\sin \phi \cos \theta \sin \beta \cos \lambda + \sin \phi \sin \theta \sin \beta \sin \lambda + \cos \phi \cos \beta) \\ &= r\rho[\sin \phi \sin \beta(\cos \theta \cos \lambda + \sin \theta \sin \lambda) + \cos \phi \cos \beta] \\ &= r\rho[\sin \phi \sin \beta \cos(\theta - \lambda) + \cos \phi \cos \beta] \end{aligned} \quad (4.34)$$

And, Eq. (4.31) can be written as

$$\cos \gamma = \cos \phi \cos \beta + \sin \phi \sin \beta \cos(\theta - \lambda) \quad (4.35)$$

Returning to the magnitude of the relative positions

$$|\mathbf{r} - \boldsymbol{\rho}| = (r^2 + \rho^2 - 2r\rho \cos \gamma)^{1/2} \quad (4.36)$$

the inverse of the value becomes

$$|\mathbf{r} - \boldsymbol{\rho}|^{-1} = \frac{1}{r} \left( 1 + \frac{\rho^2}{r^2} - 2\frac{\rho}{r} \cos \gamma \right)^{-1/2} \quad (4.37)$$

The terms in the parentheses on the right-hand side of this equation can be considered small compared to one. Using the following series expansion for the  $n$ th power

$$\begin{aligned} (1 \pm x)^{-n} &= 1 \mp nx + \frac{n(n+1)}{2!} x^2 \mp \frac{n(n+1)(n+2)}{3!} x^3 + \dots \\ \forall x^2 < 1 \end{aligned} \quad (4.38)$$

and the small variable defined as

$$x \equiv \frac{\rho^2}{r^2} - 2\frac{\rho}{r}\cos\gamma \quad (4.39)$$

then, using Eqs. (4.38) and (4.39), Eq. (4.37) can be expanded as

$$\begin{aligned} |r - \rho|^{-1} &= \frac{1}{r} \left[ 1 - \frac{1}{2} \left( \frac{\rho^2}{r^2} - 2\frac{\rho}{r}\cos\gamma \right) \right. \\ &\quad \left. + \frac{3}{8} \left( \frac{\rho^2}{r^2} - 2\frac{\rho}{r}\cos\gamma \right)^2 \right. \\ &\quad \left. - \frac{5}{16} \left( \frac{\rho^2}{r^2} - 2\frac{\rho}{r}\cos\gamma \right)^3 + \dots \right] \\ &= \frac{1}{r} \left[ 1 + \frac{\rho}{r}\cos\gamma + \frac{1}{2} \frac{\rho^2}{r^2} (3\cos^2\gamma - 1) \right. \\ &\quad \left. + \frac{1}{2} \frac{\rho^3}{r^3} (5\cos^3\gamma - 3\cos\gamma) \right. \\ &\quad \left. + \frac{1}{8} \frac{\rho^4}{r^4} (35\cos^4\gamma - 30\cos^2\gamma + 3) + \dots \right] \\ &\equiv \frac{1}{r} \left[ 1 + \frac{\rho}{r} P_1(\cos\gamma) + \frac{\rho^2}{r^2} P_2(\cos\gamma) \right. \\ &\quad \left. + \frac{\rho^3}{r^3} P_3(\cos\gamma) \right. \\ &\quad \left. + \frac{\rho^4}{r^4} P_4(\cos\gamma) + \dots \right] \\ &= \frac{1}{r} \sum_{k=0}^{\infty} \left( \frac{\rho}{r} \right)^k P_k(\cos\gamma) \end{aligned} \quad (4.40)$$

where the following definitions for Legendre polynomials have been used:

$$\begin{aligned} P_1(\cos\gamma) &\equiv \cos\gamma \\ P_2(\cos\gamma) &\equiv \frac{1}{2}(3\cos^2\gamma - 1) \\ P_3(\cos\gamma) &\equiv \frac{1}{2}(5\cos^3\gamma - 3\cos\gamma) \\ P_4(\cos\gamma) &\equiv \frac{1}{8}(35\cos^4\gamma - 30\cos^2\gamma + 3) \end{aligned} \quad (4.41)$$

etc.

Integrating the differential potential function in Eq. (4.29) over the range of variables encompassing the mass, the gravitational potential becomes

$$\begin{aligned}
 U &= \frac{G}{r} \iiint \sum_{k=0}^{\infty} \left(\frac{\rho}{r}\right)^k P_k(\cos \gamma) dm \\
 &= \frac{G}{r} \iiint dm + \frac{G}{r} \iiint \frac{\rho}{r} P_1(\cos \gamma) dm \\
 &\quad + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right)^2 P_2(\cos \gamma) dm + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right)^3 P_3(\cos \gamma) dm + \dots \\
 &= \frac{GM}{r} + \frac{G}{r^2} \iiint \rho \cos \gamma dm + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right)^2 P_2(\cos \gamma) dm \\
 &\quad + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right)^3 P_3(\cos \gamma) dm + \dots
 \end{aligned} \tag{4.42}$$

To exploit their properties, the following Legendre polynomial derivative relationship is used:

$$P_k^j(v) = (1 - v^2)^{1/2} \frac{d^j}{dv^j} P_k(v) \tag{4.43}$$

Defining the variable  $v$  as

$$v \equiv \cos \phi \tag{4.44}$$

then, the derivative polynomials become

$$\begin{aligned}
 P_1^1 &= (1 - \cos^2 \phi)^{1/2} \frac{d}{d\phi} v = \sin \phi \\
 P_2^1 &= (1 - \cos^2 \phi)^{1/2} \frac{d}{d\phi} \left[ \frac{1}{2}(3v^2 - 1) \right] = 3 \sin \phi \cos \phi \\
 P_2^2 &= (1 - \cos^2 \phi) \frac{d^2}{d\phi^2} \left[ \frac{1}{2}(3v^2 - 1) \right] = 3 \sin^2 \phi
 \end{aligned} \tag{4.45}$$

etc. Using these expressions, the gravitational potential in Eq. (4.42) is written as

$$\begin{aligned}
 U &= \frac{GM}{r} + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right) P_1(\cos \gamma) dm + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right)^2 P_2(\cos \gamma) dm + \dots \\
 &= \frac{GM}{r} + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right) [P_1(\cos \phi) P_1(\cos \beta) + P_1^1(\cos \phi) P_1^1(\cos \beta) \cos(\theta - \lambda)] dm \\
 &\quad + \frac{G}{r} \iiint \left(\frac{\rho}{r}\right)^2 \left[ P_2(\cos \phi) P_2(\cos \beta) + \frac{1}{3} P_2^1(\cos \phi) P_2^1(\cos \beta) \cos(\theta - \lambda) \right. \\
 &\quad \left. + \frac{1}{12} P_2^2(\cos \phi) P_2^2(\cos \beta) \cos 2(\theta - \lambda) \right] dm + \dots
 \end{aligned} \tag{4.46}$$

Factoring terms with the  $p$  positions'  $\phi$  dependency out of the mass integrals

$$\begin{aligned}
 U = & \frac{GM}{r} + \frac{G}{r^2} \iiint \rho P_1(\cos \beta) dm P_1(\cos \phi) \\
 & + \frac{G}{r^2} \iiint \rho P_1^1(\cos \beta) \cos(\theta - \lambda) dm P_1^1(\cos \phi) \\
 & + \frac{G}{r^3} \iiint \rho^2 P_2(\cos \beta) dm P_2(\cos \phi) \\
 & + \frac{G}{r^3} \iiint \rho^2 \frac{1}{3} P_2^1(\cos \beta) \cos(\theta - \lambda) dm P_2^1(\cos \phi) \\
 & + \frac{G}{r^3} \iiint \rho^2 \frac{1}{12} P_2^2(\cos \beta) \cos 2(\theta - \lambda) dm P_2^2(\cos \phi) + \dots
 \end{aligned} \quad (4.47)$$

Expanding the cosine of the difference of the two angles, this equation is rewritten as

$$\begin{aligned}
 U = & \frac{GM}{r} + \frac{G}{r^2} \iiint \rho P_1(\cos \beta) dm P_1(\cos \phi) \\
 & + \frac{G}{r^2} \iiint \rho P_1^1(\cos \beta) \cos \lambda dm P_1^1(\cos \phi) \cos \theta \\
 & + \frac{G}{r^2} \iiint \rho P_1^1(\cos \beta) \sin \lambda dm P_1^1(\cos \phi) \sin \theta \\
 & + \frac{G}{r^3} \iiint \rho^2 P_2(\cos \beta) dm P_2(\cos \phi) dm \\
 & + \frac{G}{r^3} \iiint \rho^2 \frac{1}{3} P_2^1(\cos \beta) \cos \lambda dm P_2^1(\cos \phi) \cos \theta \\
 & + \frac{G}{r^3} \iiint \rho^2 \frac{1}{3} P_2^1(\cos \beta) \sin \lambda dm P_2^1(\cos \phi) \sin \theta \\
 & + \frac{G}{r^3} \iiint \rho^2 \frac{1}{12} P_2^2(\cos \beta) \cos 2\lambda dm P_2^2(\cos \phi) \cos 2\theta \\
 & + \frac{G}{r^3} \iiint \rho^2 \frac{1}{12} P_2^2(\cos \beta) \sin 2\lambda dm P_2^2(\cos \phi) \sin 2\theta + \dots
 \end{aligned} \quad (4.48)$$

Or, to summarize this equation in a compact form, the gravitational potential becomes

$$\begin{aligned}
 U \equiv & \frac{GM}{r} + \sum_{k=1}^{\infty} \frac{1}{r^{k+1}} \\
 & \times \left\{ A_k P_k(\cos \phi) + \sum_{j=1}^k \left[ B_k^j P_k^j(\cos \phi) \cos j\theta + C_k^j P_k^j(\cos \phi) \sin j\theta \right] \right\}
 \end{aligned} \quad (4.49)$$

where the elemental mass

$$dm \equiv D(\rho, \beta, \lambda) \rho^2 \sin \beta d\rho d\beta d\lambda \quad (4.50)$$

and other coefficients in the summations are defined as

$$A_k \equiv G \iiint \rho^{k+2} D(\rho, \beta, \lambda) P_k(\cos \beta) \sin \beta d\rho d\beta d\lambda \quad (4.51)$$

$$B_k^j \equiv 2G \frac{(k-j)}{(k+j)} \iiint \rho^{k+2} D(\rho, \beta, \lambda) P_k^j(\cos \beta) \cos j\lambda \sin \beta d\rho d\beta d\lambda \quad (4.52)$$

$$C_k^j \equiv 2G \frac{(k-j)}{(k+j)} \iiint \rho^{k+2} D(\rho, \beta, \lambda) P_k^j(\cos \beta) \sin j\lambda \sin \beta d\rho d\beta d\lambda \quad (4.53)$$

If it is assumed that the Earth possesses axial symmetry about the vertical axis, then

$$D(\rho, \beta, \lambda) \equiv D(\rho, \beta) \quad (4.54)$$

and the integrals in Eqs (4.52) and (4.53) equate to zero as

$$\int_0^{2\pi} \sin j\lambda d\lambda = \int_0^{2\pi} \cos j\lambda d\lambda = 0 \Rightarrow B_k^j = C_k^j = 0 \quad \forall j \quad (4.55)$$

Also, if it is assumed that the axis of symmetry passes through the center of the Earth's mass, then

$$\frac{G}{r^2} \iiint \rho \cos \gamma dm = 0 \quad (4.56)$$

With these assumptions, the gravitational potential simplifies to

$$U = \frac{\mu}{r} + \sum_{k=2}^{\infty} \frac{A_k}{r^{k+1}} P_k(\cos \phi) \quad (4.57)$$

where, for notional simplification,  $GM \equiv \mu$ .

#### 4.2.2 Gravitational Acceleration

The potential function in Eq. (4.57) is rewritten, including only the first nonzero coefficient (see Problem 4.9) and defining the coefficient  $A_2$  in terms of  $J_2$ .

$$U_{J_2}(r, \varphi_c) = \frac{\mu}{r} \left[ 1 - J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} (3 \sin^2 \varphi_c - 1) \right] \quad (4.58)$$

where, for a point on the surface,

$$r^2 = \beta^2 + z^2$$

and

$$\sin \varphi_c = \frac{z}{r}$$

Then, the potential becomes

$$U_{J_2} = \frac{\mu}{r} \left\{ 1 - J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{z}{r} \right)^2 - 1 \right] \right\} \quad (4.59)$$

The gradient of this potential, with respect to  $\beta$  and  $z$ , yields the gravitational acceleration in those directions:

$$\begin{aligned} G_\beta &\equiv \frac{\partial U}{\partial \beta} = -\frac{\beta}{r^3} \mu \left\{ 1 - J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{z}{r} \right)^2 - 1 \right] \right\} \\ &\quad + \frac{\mu}{r} \left\{ +J_2 2 \frac{\beta}{r^2} \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{z}{r} \right)^2 - 1 \right] + J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 2 \frac{\beta}{r^2} 3 \left( \frac{z}{r} \right)^2 \right] \right\} \\ &= -\frac{\mu}{r^2} \left\{ 1 - J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{z}{r} \right)^2 - 1 \right] - 2 J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{z}{r} \right)^2 - 1 \right] \right. \\ &\quad \left. - J_2 \left( \frac{r_e}{r} \right)^2 \left[ 3 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{\beta}{r} \\ &= -\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 1 - 3 \left( \frac{z}{r} \right)^2 \right] - \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 2 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{\beta}{r} \\ &= -\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 1 - 5 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{\beta}{r} \end{aligned} \quad (4.60)$$

$$\begin{aligned} G_z &\equiv \frac{\partial U}{\partial z} = -\frac{z}{r^3} \mu \left\{ 1 - J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{z}{r} \right)^2 - 1 \right] \right\} \\ &\quad + \frac{\mu}{r} \left\{ +J_2 2 \frac{z}{r^2} \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{z}{r} \right)^2 - 1 \right] - J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 3 \left( \frac{2z}{r^2} - \frac{2z^3}{r^4} \right) \right] \right\} \\ &= -\frac{\mu}{r^2} \left\{ 1 + J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 1 - 3 \left( \frac{z}{r} \right)^2 \right] + 2 J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} \left[ 1 - 3 \left( \frac{z}{r} \right)^2 \right] \right. \\ &\quad \left. + J_2 \left( \frac{r_e}{r} \right)^2 \frac{3}{2} \left[ 2 - 2 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{z}{r} \\ &= -\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 1 - 3 \left( \frac{z}{r} \right)^2 \right] + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 2 - 2 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{z}{r} \\ &= -\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 3 - 5 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{z}{r} \end{aligned} \quad (4.61)$$

The component in the  $\beta$  is resolved into  $x$  and  $y$  components to yield the gravitational accelerations in Earth-centered coordinates as

$$G_x = -\frac{\mu}{R^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{R} \right)^2 \left[ 1 - 5 \left( \frac{z}{R} \right)^2 \right] \right\} \frac{x}{R} + \dots \quad (4.62)$$

$$G_y = -\frac{\mu}{R^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{R} \right)^2 \left[ 1 - 5 \left( \frac{z}{R} \right)^2 \right] \right\} \frac{y}{R} + \dots \quad (4.63)$$

$$G_z = -\frac{\mu}{R^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{R} \right)^2 \left[ 3 - 5 \left( \frac{z}{R} \right)^2 \right] \right\} \frac{z}{R} + \dots \quad (4.64)$$

where the radial position to the location in Earth-centered coordinates is

$$R = \sqrt{x^2 + y^2 + z^2} \quad (4.65)$$

These equations are approximate with only the first two terms,  $\mu$  and  $J_2$ , included. The  $J_2$  term is the first nonspherical model term included in the gravitational model, and its contribution to the model is to include effects caused by the Earth's oblateness.

#### 4.2.3 Gravity

Gravity is the acceleration on a rotating Earth. The preceding gravitation acceleration model is referenced to non-rotating frame. This acceleration is adjusted to account for the centripetal acceleration caused by the Earth's rotation. Including this acceleration, the gravity vector is

$$\mathbf{g} = \mathbf{G} - \Omega_{i/e} \Omega_{i/e} \mathbf{r} \quad (4.66)$$

*WGS gravity model.* The magnitude of gravity at the surface of the WGS-84 ellipsoid can be expressed in the following form [4]:

$$g = g_{\text{WGS}_0} \frac{(1 + g_{\text{WGS}_1} \sin^2 \phi)}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}} \quad (4.67)$$

This simple mathematical form to describe gravity motivates the use of local-level referenced navigation state equations, which are much simpler than alternatives, that is, ECEF navigation state equations.

Gravity is affected by altitude (the inverse square law of acceleration). The variation of gravity with altitude is included in the model form in the preceding equation by the additional term  $\sim 2(h/R)$ .

**Gravity anomalies.** If the Earth conformed to a homogenous ellipsoid, the local gravity vector direction would be normal to the ellipsoid surface (see Problem 4.10). Gravity anomalies are deviations from the gravity model as maintained in the navigation system. An analytical gravity model is limited in that it cannot include the many possible variations. Higher-order models are used in special applications, that is, precision navigation systems; however, most terrestrial navigation systems' performance goals can be satisfied by models that include only the few terms already discussed.

Gravity deviations are represented by the following gravity vector referenced in the local-level geographic frame [1]:

$$\mathbf{g}^g = \begin{bmatrix} \xi g \\ -\eta g \\ g \end{bmatrix} \quad (4.68)$$

where  $\xi$  is the meridian deflection of the vertical (positive about east),  $\eta$  the normal deflection of the vertical (positive about north), and  $g$  the gravity magnitude. The gravity deflections just given can be several arc seconds in magnitude. These deflections result in tens of micro gs of unmodeled acceleration. Both  $\xi$  and  $\eta$  are relatively constant over short distances.

### 4.3 Chapter Summary

Models for the Earth's geometrical shape were presented, and equations for converting latitude, longitude, and altitude position into ECEF positions and the reverse were developed. These conversions are used in the global-positioning-system case study presented later in Chapter 12. Expressions for Earth radii of curvature were developed for use in the numerical integration of navigation state equations that describe vehicle motion (see Sec. 5.2). Simplified gravity models based on WGS-84 definitions were presented. These models describe gravity at the surface of an ellipsoid defined as nominally normal to that ellipsoid. This simple mathematical form to describe gravity shows that the use of local-level referenced navigation state equations is much simpler than alternatives, that is, ECEF navigation state equations.

Approximations to shape and gravity models developed in this chapter are used later in the linearization of navigation state equations (see Sec. 5.3).

### Problems

- 4.1** This exercise demonstrates the sensitivity of  $R_{\text{normal}}$  to latitude. From Eq. (4.28), obtain the following:

$$\frac{\partial R_{\text{normal}}}{\partial \phi} = \frac{R_{\text{normal}} s \phi c \phi \varepsilon^2}{(1 - \varepsilon^2 s^2 \phi)}$$

then

$$\frac{\delta R_{\text{normal}}}{R_{\text{normal}}} \propto \varepsilon^2 \delta \phi$$

to demonstrate that the sensitivity of  $R_{\text{normal}}$  to latitude is of order  $\varepsilon^2$ , a small quantity. Show the same sensitivity of  $R_{\text{meridian}}$ .

- 4.2** This exercise demonstrates the iteration method to convert ECEF positions to latitude/longitude/altitude. The Newton iteration algorithm [5] for finding the root of the continuously differentiable nonlinear function  $f(x)$  can be stated as

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad \text{for } i = 0, \dots, n$$

until the change in the computed root  $x_{i+1}$  is small compared to the previous value.

Establish the iteration function to find the latitude

$$f(\phi) = z_{\text{ECEF}} - (x_{\text{ECEF}}^2 + y_{\text{ECEF}}^2)^{1/2} t\phi + \frac{r_e \varepsilon^2 s\phi}{(1 - \varepsilon^2 s^2 \phi)^{1/2}} = 0$$

whose derivative with respect to latitude  $\phi$  becomes

$$f'(\phi) = -\frac{(x_{\text{ECEF}}^2 + y_{\text{ECEF}}^2)^{1/2}}{c^2 \phi} + \frac{r_e \varepsilon^2 c \phi}{(1 - \varepsilon^2 s^2 \phi)^{3/2}}$$

and use the algorithm form

$$\phi_{i+1} = \phi_i - \frac{f(\phi_i)}{f'(\phi_i)}$$

- 4.3** This exercise uses the substitution method to convert ECEF to latitude/longitude/altitude. Using the results of Sec. 4.1, demonstrate the following relationships:

$$h = \sqrt{x_{\text{ECEF}}^2 + y_{\text{ECEF}}^2} \frac{1}{\cos \phi} - \frac{r_e}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}}$$

From Eq. (4.11),

$$\tan \phi_c = \frac{z_{\text{ECEF}}}{(x_{\text{ECEF}}^2 + y_{\text{ECEF}}^2)^{1/2}}$$

Obtain the following equation for the geodetic latitude:

$$\tan \phi = \tan \phi_c \left\{ \frac{[r_e/(1 - \varepsilon^2 \sin^2 \phi)^{1/2} + h]}{[r_e(1 - \varepsilon^2)/(1 - \varepsilon^2 \sin^2 \phi)^{1/2} + h]} \right\}$$

Or redefine terms to obtain the following:

$$\tan \phi = \tan \phi_c \left\{ \frac{[R + h]}{[R(1 - \varepsilon^2) + h]} \right\}$$

where

$$R \equiv \frac{r_e}{(1 - \varepsilon^2 \sin^2 \phi)^{1/2}}$$

Assume a two-step procedure to solve for geodetic latitude. First, approximate the values for

$$R \approx r_e$$

and

$$h = 0$$

Using these values in the second equation for  $\tan \phi$ , compute a first approximation to the geodetic latitude, and, with that result in the equation for  $h$ , compute a first approximation to altitude. Using these results for geodetic latitude and altitude, recompute these values using 1) the same equation for  $\tan \phi$  (but now using the equation for  $R$  to obtain a refined value for geodetic latitude) and 2) the same equation for  $h$  to also obtain a refined value for altitude.

Finally, to complete the Earth surface positions, longitude is computed from Eqs. (4.20) and (4.21):

$$\lambda = \tan^{-1} \left( \frac{y^{ECEF}}{x^{ECEF}} \right)$$

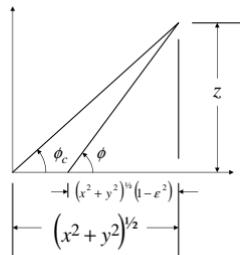
This value of longitude is used with the results of the iteration for latitude and altitude.

- 4.4** Analytical substitution is used in this exercise to obtain a first-order solution for geodetic latitude and altitude accounting for the Earth's oblateness. Consider the following figure showing the geocentric latitude and its

first-order correction indicated. From this figure, obtain the following equations:

$$\tan \phi_1 = \frac{z}{(x^2 + y^2)^{1/2}(1 - \varepsilon^2)}$$

$$\sin \phi_1 = \frac{z}{[(x^2 + y^2)(1 - \varepsilon^2)^2 + z^2]^{1/2}}$$



and

$$\cos \phi_1 = \frac{(x^2 + y^2)^{1/2}(1 - \varepsilon^2)}{[(x^2 + y^2)(1 - \varepsilon^2)^2 + z^2]^{1/2}}$$

Substitute these equations into the equation for altitude to obtain its first-order solution:

$$h_1 = \left[ 1 + \frac{z^2}{(x^2 + y^2)(1 - \varepsilon^2)} \right]^{1/2} \times \left\{ (x^2 + y^2)^{1/2} - r_e / \left[ 1 + \frac{z^2}{(x^2 + y^2)(1 - \varepsilon^2)} \right]^{1/2} \right\}$$

- 4.5** This exercise forms  $\omega_{e/n}^n$  for a wander-azimuth frame. Consider the wander-azimuth frame presented in Fig. 3.6. The horizontal components of the geographic frame referenced rates are

$$\rho_{\text{north}} = -\frac{v_{\text{west}}}{R_{\text{normal}} + h}$$

$$\rho_{\text{west}} = \frac{v_{\text{north}}}{R_{\text{meridian}} + h}$$

Use the transformation

$$\begin{bmatrix} \rho_x \\ \rho_y \end{bmatrix} = \begin{bmatrix} c\alpha & s\alpha \\ -s\alpha & c\alpha \end{bmatrix} \begin{bmatrix} \rho_{\text{north}} \\ \rho_{\text{west}} \end{bmatrix}$$

from the geographic frame to the wander-azimuth frame to obtain the following rate components:

$$\rho_x = -v_y \left( \frac{c^2 \alpha}{R_n + h} + \frac{s^2 \alpha}{R_m + h} \right) + v_x \left( \frac{1}{R_m + h} - \frac{1}{R_n + h} \right) s \alpha c \alpha$$

$$\rho_y = v_x \left( \frac{c^2 \alpha}{R_m + h} + \frac{s^2 \alpha}{R_n + h} \right) + v_y \left( \frac{1}{R_n + h} - \frac{1}{R_m + h} \right) s \alpha c \alpha$$

where

$$\begin{bmatrix} v_{\text{north}} \\ v_{\text{west}} \end{bmatrix} = \begin{bmatrix} c\alpha & -s\alpha \\ s\alpha & c\alpha \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

- 4.6** This exercise demonstrates approximations to gravitational variation. Using the results of Sec. 4.2, make the assumption that the  $J_2$  term is negligible to obtain the following:

$$\frac{\partial \mathbf{G}}{\partial \mathbf{r}} = \omega_s^2 (3\mathbf{e}\mathbf{e}^T - I)$$

where

$$\omega_s^2 = \frac{\mu}{R^3}$$

and the position unit vector is

$$\mathbf{e} = \frac{1}{R} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Note that gravity can be approximated as  $g \approx \mu/R^2$  and the preceding frequency becomes  $\omega_s^2 = g/R^2$ —the Schuler frequency (see Example 5.1).

- 4.7** This exercise shows the variation of gravity with altitude. In Sec. 4.2, it was indicated that gravity varied with altitude. Assume a spherical Earth and the following simplified inverse square law gravity model:

$$g = g_0 \left( \frac{R}{R+h} \right)^2$$

where  $g_0$  is the gravity magnitude at zero altitude  $h$ . Take the partial derivative of this equation with respect to altitude to obtain

$$\frac{\partial g}{\partial h} = -2 \left( \frac{g}{R+h} \right)$$

- 4.8** This exercise demonstrates navigation frame position error induced Earth rate error. For most terrestrial navigation applications, Earth rate is assumed constant and known. As will be shown later, Earth rate referenced in the local navigation frame is used in formulating navigation state equations. This transformed Earth rate can contain error if the local position is in error. Form the computed navigation frame referenced Earth rate by

transforming the known Earth rate with a computed Earth-to-navigation frame direction cosine matrix as

$$\begin{aligned}\bar{\boldsymbol{\omega}}_{i/e}^n &= \bar{C}_e^n \boldsymbol{\omega}_{i/e}^e \\ &= [I - (\delta\theta \times)] C_e^n \boldsymbol{\omega}_{i/e}^e\end{aligned}$$

Show that the resulting navigation frame Earth rate error becomes

$$\delta\boldsymbol{\omega}_{i/e}^n \equiv \bar{\boldsymbol{\omega}}_{i/e}^n - \boldsymbol{\omega}_{i/e}^n = -(\delta\theta \times) \boldsymbol{\omega}_{i/e}^n$$

- 4.9** Develop an expression for the gravitational potential assuming the Earth possesses a homogeneous density

$$D(\rho, \beta, \lambda) \equiv D$$

Show for the ellipsoid shape

$$\begin{aligned}\frac{x^2}{a^2} + \frac{z^2}{b^2} &= 1 \quad \Rightarrow \quad b^2 x^2 + a^2 z^2 = a^2 b^2 \\ x = r \sin \beta &\quad \Rightarrow \quad x^2 = r^2 \sin^2 \beta \\ z = r \cos \beta &\quad \Rightarrow \quad z^2 = r^2 \cos^2 \beta \\ r^2 &= \frac{a^2 b^2}{b^2 \sin^2 \beta + a^2 \cos^2 \beta}\end{aligned}$$

Then, using this expression, show that the mass integral becomes

$$\begin{aligned}M &\equiv \iiint D(\rho, \beta, \lambda) \rho^2 \sin \beta d\rho d\beta d\lambda \\ &= 2\pi D \iint \rho^2 \sin \beta d\rho d\beta \\ &= 2\pi D \int \left[ \frac{1}{3} \rho^3 \right]_0^{r(\beta)} \sin \beta d\beta \\ &= \frac{2}{3} \pi a^3 b^3 D \int_0^\pi \frac{\sin \beta}{(b^2 \sin^2 \beta + a^2 \cos^2 \beta)^{3/2}} d\beta \\ &= \frac{2}{3} \pi a^3 b^3 D \left( \frac{2}{ab^2} \right) \\ &= \frac{4}{3} \pi a^2 b D\end{aligned}$$

and, the first coefficient in Eq. (4.57) equates to zero

$$\begin{aligned}
 A_1 &\equiv G \iiint \rho^3 D P_1(\cos \beta) \sin \beta d\rho d\beta d\lambda \\
 &= 2\pi DG \iint \rho^3 \sin \beta \cos \beta d\rho d\beta \\
 &= 2\pi DG \int \left[ \frac{1}{4} \rho^4 \right]_0^{r(\beta)} \sin \beta \cos \beta d\beta \\
 &= \frac{1}{2} \pi a^4 b^4 DG \int_0^\pi \frac{\sin \beta \cos \beta}{(b^2 \sin^2 \beta + a^2 \cos^2 \beta)^2} d\beta \\
 &= \frac{1}{2} \pi a^4 b^4 DG \int_0^\pi \frac{d}{d\beta} \left[ \frac{1}{2(a^2 - b^2)} \frac{1}{(b^2 \sin^2 \beta + a^2 \cos^2 \beta)} \right] d\beta \\
 &= \frac{1}{2} \pi a^4 b^4 DG \frac{1}{2(a^2 - b^2)} \left[ \frac{1}{a^2} - \frac{1}{a^2} \right] \\
 &= 0
 \end{aligned}$$

Show that the first non-zero coefficient in the gravitational potential equation becomes

$$\begin{aligned}
 A_2 &\equiv G \iiint \rho^4 D P_2(\cos \beta) \sin \beta d\rho d\beta d\lambda \\
 &= 2\pi DG \iint \rho^4 \frac{1}{2} (3 \cos^2 \beta - 1) \sin \beta d\rho d\beta \\
 &= \pi DG \int \left[ \frac{1}{5} \rho^5 \right]_0^{r(\beta)} (3 \cos^2 \beta - 1) \sin \beta d\beta \\
 &= \frac{1}{5} \pi a^5 b^5 DG \int_0^\pi \frac{(3 \cos^2 \beta - 1) \sin \beta}{(b^2 \sin^2 \beta + a^2 \cos^2 \beta)^{5/2}} d\beta \\
 &= \frac{1}{5} \pi a^5 b^5 DG \left[ -\frac{4}{3} \frac{(a^2 - b^2)}{a^3 b^4} \right] \\
 &= -\frac{4}{3} \pi a^2 b DG \frac{1}{5} (a^2 - b^2) \\
 &= -MG \frac{1}{5} (a^2 - b^2)
 \end{aligned}$$

Defining the following

$$\varepsilon^2 = 1 - \frac{b^2}{a^2}$$

$$r^2 = x^2 + y^2 + z^2$$

$$\cos \phi = \frac{z}{r}$$

and showing the potential, including terms up to the second coefficient, become

$$\begin{aligned} U_{k=2} &= \frac{GM}{r} + \frac{A_2}{r^3} P_2(\cos \phi) \\ &= \frac{GM}{r} + \frac{1}{r^3} \left[ -GM \frac{1}{5}(a^2 - b^2) \right] \frac{1}{2} (3 \cos^2 \phi - 1) \\ &= \frac{GM}{r} + \frac{GM}{10r^3} \left[ -a^2 \left( 1 - \frac{b^2}{a^2} \right) \right] \left( \frac{3z^2 - r^2}{r^2} \right) \\ &= \frac{GM}{r} \left[ 1 + \frac{a^2}{10r^4} (x^2 + y^2 - 2z^2) \varepsilon^2 \right] \end{aligned}$$

- 4.10** This exercise completes steps between the first and second lines in Eq. (4.46). Recall the expression from Eq. (4.35):

$$\cos \gamma = \cos \phi \cos \beta + \sin \phi \sin \beta \cos (\theta - \lambda)$$

Because

$$P_1(\cos \nu) = \cos \nu$$

and, from Eqs (4.45)

$$P_1^1(\cos \nu) = \sin \nu$$

then the first term in the second line in Eq. (4.46) becomes

$$P_1(\cos \gamma) = P_1(\cos \phi)P_1(\cos \beta) + P_1^1(\cos \phi)P_1^1(\cos \beta) \cos (\theta - \lambda)$$

Continue, and demonstrate that the second term in this equation can be written as

$$\begin{aligned}
 P_2(\cos \gamma) &= \frac{1}{2}(3 \cos^2 \gamma - 1) \\
 &= \frac{1}{2}(3 \cos^2 \phi \cos^2 \beta + \frac{3}{2} \sin^2 \phi \sin^2 \beta - 1) \\
 &\quad + 3 \sin \phi \cos \phi \sin \beta \cos \beta \cos(\theta - \lambda) \\
 &\quad + \frac{3}{4} \sin^2 \phi \sin^2 \beta \cos 2(\theta - \lambda) \\
 &= \frac{1}{2} \left( 3 \cos^2 \phi \cos^2 \beta + \frac{3}{2} \sin^2 \phi \sin^2 \beta - 1 \right) \\
 &\quad + \frac{1}{3}(3 \sin \phi \cos \phi)(3 \sin \beta \cos \beta) \cos(\theta - \lambda) \\
 &\quad + \frac{1}{12}(3 \sin^2 \phi)(3 \sin^2 \beta) \cos 2(\theta - \lambda) \\
 &= P_2(\cos \phi)P_2(\cos \beta) \\
 &\quad + \frac{1}{3}P_2^1(\cos \phi)P_2^1(\cos \beta) \cos(\theta - \lambda) \\
 &\quad + \frac{1}{12}P_2^2(\cos \phi)P_2^2(\cos \beta) \cos 2(\theta - \lambda)
 \end{aligned}$$

- 4.11.** This exercise examines the relationship between the surface normal and direction of gravity for a oblate spheroid. Consider the equation of an ellipsoid surface with axes defined in Fig. 4.1.

$$s(\beta, z) = \frac{\beta^2}{r_e^2} + \frac{z^2}{r_p^2} - 1$$

Components of the surface gradient vector are computed as

$$n \equiv \frac{\nabla s}{|\nabla s|} = \frac{r_e(1 - \varepsilon^2)^{1/2}}{[1 - \varepsilon^2(\beta/r_e)^2]^{1/2}} \begin{bmatrix} \frac{\beta}{r_e^2} \\ \frac{z}{r_e^2(1 - \varepsilon^2)} \end{bmatrix}$$

where  $\varepsilon$  is defined in Eq. (4.1). An angle is defined as the arc tangent of the ratio of the  $\beta$  and  $z$  components of this vector.

$$\gamma \equiv \tan^{-1} \left[ \frac{(1 - \varepsilon^2)\beta}{z} \right]$$

Now, consider the gravitation acceleration from Eqs. (4.60) and (4.61) and only including terms up to  $J_2$ .

$$G_\beta = -\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 1 - 5 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{\beta}{r}$$

$$G_z = -\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 3 - 5 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{z}{r}$$

From these gravitational components, gravity components are written as

$$g_\beta = G_\beta + \omega^2 \beta$$

$$g_z = G_z$$

Again define an angle from the  $\beta$  and  $z$  components of gravity as was just done with the surface normal

$$\delta \equiv \tan^{-1} \left( \frac{g_\beta}{g_z} \right)$$

Substituting from the preceding,

$$\begin{aligned} \delta &= \tan^{-1} \left( \frac{-\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 1 - 5 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{\beta}{r} + \omega^2 \beta}{-\frac{\mu}{r^2} \left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 3 - 5 \left( \frac{z}{r} \right)^2 \right] \right\} \frac{z}{r}} \right) \\ &= \tan^{-1} \left( \frac{\left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 1 - 5 \left( \frac{z}{r} \right)^2 \right] - \frac{\omega^2 r^3}{\mu} \right\} \beta}{\left\{ 1 + \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \left[ 3 - 5 \left( \frac{z}{r} \right)^2 \right] \right\} z} \right) \\ &\equiv \tan^{-1} \left[ \frac{(1 + \varepsilon_1) \beta}{(1 + \varepsilon_2) z} \right] \\ &\approx \tan^{-1} \left[ (1 + \varepsilon_1)(1 - \varepsilon_2) \frac{\beta}{z} \right] \\ &\approx \tan^{-1} \left\{ [1 + (\varepsilon_1 - \varepsilon_2)] \frac{\beta}{z} \right\} \end{aligned}$$

Using WGS-84 Earth model parameters in Table 4.2 for shape and gravity, show that the difference between the angles  $\varepsilon^2$  and  $-(\varepsilon_1 - \varepsilon_2)$  is less than  $10^{-5}$  radians.

Note: the  $\varepsilon$ 's in the fourth and last rows in the preceding development are assumed to be small and defined as

$$\begin{aligned}\varepsilon_1 &= \frac{3}{2} J_2 \left(\frac{r_e}{r}\right)^2 \left[ 1 - 5 \left(\frac{z}{r}\right)^2 \right] - \frac{\omega^2 r^3}{\mu} \\ \varepsilon_2 &= \frac{3}{2} J_2 \left(\frac{r_e}{r}\right)^2 \left[ 3 - 5 \left(\frac{z}{r}\right)^2 \right]\end{aligned}$$

Then, assuming  $r \approx r_e$ , the following is obtained for the difference in  $\varepsilon$ 's.

$$-(\varepsilon_1 - \varepsilon_2) = 3J_2 + \frac{\omega^2 r_e^3}{\mu}$$

Therefore, the surface normal and gravity directions, predicted from the preceding equations that include terms only up to  $J_2$  in the gravitational model, are in very close agreement.

- 4.12** An alternate form for gravity is developed in this exercise. Modify the potential in Eq. (4.58) to include Earth's rotation as

$$V_{J_2}(r, \varphi_c) = \frac{\mu}{r} \left[ 1 - J_2 \left(\frac{r_e}{r}\right)^2 \frac{1}{2} (3 \sin^2 \varphi_c - 1) + \frac{1}{2} \frac{\omega^2 r^3 \cos^2 \varphi_c}{\mu} \right]$$

Establish the following gradients:

$$\begin{aligned}g_r &\equiv \frac{\partial V}{\partial r} \\ &= -\frac{\mu}{r^2} \left[ 1 - J_2 \left(\frac{r_e}{r}\right)^2 \frac{1}{2} (3 \sin^2 \varphi_c - 1) + \frac{1}{2} \frac{\omega^2 r^3 \cos^2 \varphi_c}{\mu} \right] \\ &\quad + \frac{\mu}{r} \left[ +\frac{2}{r} J_2 \left(\frac{r_e}{r}\right)^2 \frac{1}{2} (3 \sin^2 \varphi_c - 1) + \frac{3}{2} \frac{\omega^2 r^2 \cos^2 \varphi_c}{\mu} \right] \\ &= -\frac{\mu}{r^2} \left[ 1 - \frac{1}{2} J_2 \left(\frac{r_e}{r}\right)^2 (3 \sin^2 \varphi_c - 1) + \frac{1}{2} \frac{\omega^2 r^3 \cos^2 \varphi_c}{\mu} \right. \\ &\quad \left. - J_2 \left(\frac{r_e}{r}\right)^2 (3 \sin^2 \varphi_c - 1) - \frac{3}{2} \frac{\omega^2 r^3 \cos^2 \varphi_c}{\mu} \right] \\ &= -\frac{\mu}{r^2} \left( 1 - \frac{3}{2} J_2 \left(\frac{r_e}{r}\right)^2 (3 \sin^2 \varphi_c - 1) - \frac{\omega^2 r^3 (1 - \sin^2 \varphi_c)}{\mu} \right] \\ &= -\frac{\mu}{r^2} \left( 1 - \frac{3}{2} J_2 \left(\frac{r_e}{r}\right)^2 - \frac{\omega^2 r^3}{\mu} - \left\{ \left[ \frac{9}{2} J_2 \left(\frac{r_e}{r}\right)^2 - \frac{\omega^2 r^3}{\mu} \right] \right\} \sin^2 \varphi_c \right)\end{aligned}$$

and

$$\begin{aligned} g_{\varphi_c} &\equiv \frac{1}{r} \frac{\partial V}{\partial \varphi_c} \\ &= \frac{\mu}{r^2} \left[ -J_2 \left( \frac{r_e}{r} \right)^2 \frac{1}{2} (3 \cdot 2 \sin \varphi_c \cos \varphi_c) - \frac{1}{2} \frac{\omega^2 r^3 2 \cos \varphi_c \sin \varphi_c}{\mu} \right] \\ &= -\frac{\mu}{r^2} \left[ 3J_2 \left( \frac{r_e}{r} \right)^2 + \frac{\omega^2 r^3}{\mu} \right] \sin \varphi_c \cos \varphi_c \end{aligned}$$

Defining the magnitude of gravity as

$$|g| = \sqrt{g_r^2 + g_{\varphi_c}^2}$$

and using the intermediate variables to ease forming the square root of their squares,

$$a \equiv \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 \quad \text{and} \quad b \equiv \frac{\omega^2 r^3}{\mu}$$

Then, rewrite these expressions as

$$\begin{aligned} g_r &= -\frac{\mu}{r} [1 - a(3 \sin^2 \varphi_c - 1) - b \cos^2 \varphi_c] \\ g_{\varphi_c} &= -\frac{\mu}{r} [2a + b] \sin \varphi_c \cos \varphi_c \end{aligned}$$

which results in the following for the gravity squared:

$$g^2 = \left( \frac{\mu}{r^2} \right)^2 \{ [1 + (a - b)] - (3a - b) \sin^2 \varphi_c \}^2$$

Returning to the original variables, show that the magnitude of gravity becomes

$$|g| = \frac{\mu}{r^2} \left( \left\{ 1 + \left[ \frac{3}{2} J_2 \left( \frac{r_e}{r} \right)^2 - \frac{\omega^2 r^3}{\mu} \right] \right\} - \left[ \frac{9}{2} J_2 \left( \frac{r_e}{r} \right)^2 - \frac{\omega^2 r^3}{\mu} \right] \sin^2 \varphi_c \right)$$

Evaluated at the equator and poles, show that the gravity magnitude becomes

$$\varphi_c = 0 \quad \Rightarrow \quad g_e = \frac{\mu}{r_e^2} \left[ 1 + \frac{3}{2} J_2 - \frac{\omega^2 r_e^2}{\mu} \right] \quad \text{at the equator}$$

$$\varphi_c = \frac{\pi}{2} \quad \Rightarrow \quad g_p = \frac{\mu}{r_p^2} \left[ 1 - 3J_2 \left( \frac{r_e}{r_p} \right)^2 \right] \quad \text{at the pole}$$



# 5

## Terrestrial Navigation

In this chapter, the following is presented for an integrated terrestrial navigation system: strap-down navigation system functional flow, local-level navigation frame mechanization equations, perturbation form of navigation system error equations, navigation system attitude error equations—psi formulation, navigation system error equations using alternative velocity error, and baro-inertial vertical channel mechanization and error equations. Inertial navigation systems for terrestrial navigation can be, and have been, mechanized in either a gimballed platform or as a strap-down unit. Most current navigation systems use strap-down sensors mechanized as a strap-down navigator. The mathematical forms for these two mechanizations differ only slightly; therefore, only equations describing a strap-down mechanization are presented in detail.

Equations describing navigation states (position, velocity, and attitude) for strap-down navigation system implementation referenced to a local-level navigation frame are developed. The resulting differential equations are nonlinear. In developing these equations, the objective is to form them in terms of sensed accelerations and rates, rather than as applied and reactionary forces and moments, as in the case of forming dynamic equations of motion. These sensed accelerations and rates are provided by inertial sensors and are described in Chapter 6.

An integrated navigation system combines navigation state data, generated by these dynamic equations, with independent redundant data in a Kalman filter algorithm. The form of the algorithm used requires a linearized error formulation of the navigation equations; with several error representations available. Many different approaches (design alternatives) are possible in forming these linearized equations, each with their advantages and disadvantages. Examples of different linearized forms are presented.

Terrestrial navigation systems stabilize a normally unstable vertical axis (channel) by using outputs from a barometric altimeter. The source of this instability is gravity variation with altitude. A typical vertical channel stabilization implementation and its associated errors are presented.

Appendix A contains additional developments for the historically significant Pinson error model.

## 5.1 Strap-Down Navigation Systems

Illustrated in Fig. 5.1 is a functional block diagram for a strap-down navigation system mechanization. In this mechanization, three accelerometers and three gyros are mounted in orthogonal triads and rigidly attached to the vehicle body. Motions sensed by the gyros, that is,  $\omega_{i/b}^b$ , are in coordinates fixed to the body. The body-referenced accelerometer outputs  $f^b$  are transformed from the body to the navigation frame in the navigation computer using the  $C_b^k$  transformation matrix.

This matrix is generated in the navigation computer by combining the rate outputs from the body-fixed gyros and the navigation frame rates created by the vehicle's movement. The function performed by the navigation computer for a strap-down system is accomplished physically for a gimballed navigation system by torquing the stabilized platform.

The gimballed navigation systems' sensors are mounted on a mechanically stabilized platform whose alignment coincides with the navigation frame. Outputs from the accelerometers  $f^k$  fixed in the navigation frame are integrated to establish velocity and position of the vehicle. Computations are performed in the navigation computer, and results from that computation are used to physically torque the stabilized platform to maintain its alignment with the navigation frame.

Each of these mechanizations has advantages and disadvantages. These generally center on cost vs performance trades. However, outputs from the two are not necessarily interchangeable. For example, body rate information, available from the strap-down navigation system, is not available or is of poor quality from the gimballed system.

Mathematical developments and equations presented in the following generally apply to both mechanizations. However, it is assumed that inertial navigation sensors are strap-down in describing the influence of these sensor errors to navigation errors.

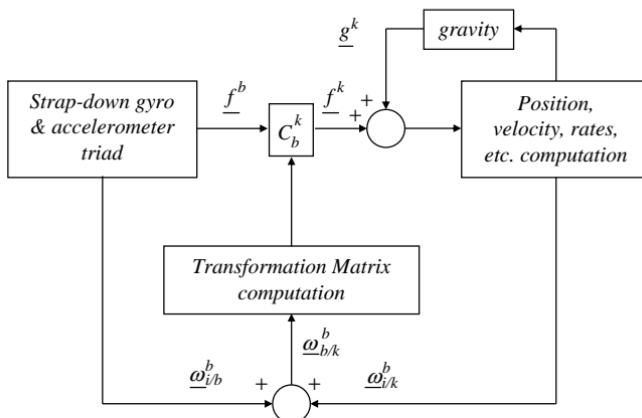


Fig. 5.1 Strap-down local-level mechanization.

## 5.2 Local-Level Navigation-Frame Mechanization Equations

Differential equations describing navigation states are developed in the following sections. In developing these equations, the objective is to form expressions for navigation states in terms of sensed accelerations and rates available from accelerometers and gyros, respectively. The resulting differential equations, one vector and two matrix, can be numerically integrated using mathematical techniques developed in Chapter 2.

### 5.2.1 Velocity Equation

The Earth relative velocity vector  $\mathbf{v}^n$  in the rotating navigation frame  $n$  is defined in terms of the rotating Earth-centered Earth-fixed frame  $e$  position as (see Problem 5.5)

$$\mathbf{v}^n \equiv C_e^n \dot{\mathbf{r}}^e \quad (5.1)$$

The time derivative of Eq. (5.1) is

$$\dot{\mathbf{v}}^n \equiv C_e^n \ddot{\mathbf{r}}^e + \dot{C}_e^n \dot{\mathbf{r}}^e \quad (5.2)$$

Each of the terms in this equation will be developed in the following. The objective for the end result is an equation that relates the accelerometer's specific force output to this rate of change of velocity in the navigation frame.

The Earth-centered Earth-fixed position vector  $\mathbf{r}^e$  is related to the nonrotating inertial frame  $i$  position as

$$\mathbf{r}^e = C_i^e \mathbf{r}^i \quad (5.3)$$

The time rate of change of this vector is

$$\dot{\mathbf{r}}^e = C_i^e \dot{\mathbf{r}}^i + \dot{C}_i^e \mathbf{r}^i \quad (5.4)$$

The time rate of change of the direction cosine matrix  $C_i^e$  is

$$\dot{C}_i^e = -C_i^e \Omega_{i/e}^i \quad (5.5)$$

which, when substituted into Eq. (5.4), yields

$$\dot{\mathbf{r}}^e = C_i^e (\dot{\mathbf{r}}^i - \Omega_{i/e}^i \mathbf{r}^i) \quad (5.6)$$

Taking the second time derivative of Eq. (5.6) results in

$$\begin{aligned} \ddot{\mathbf{r}}^e &= C_i^e (\ddot{\mathbf{r}}^i - \Omega_{i/e}^i \dot{\mathbf{r}}^i - \dot{\Omega}_{i/e}^i \mathbf{r}^i) + \dot{C}_i^e (\dot{\mathbf{r}}^i - \Omega_{i/e}^i \mathbf{r}^i) \\ &= C_i^e (\ddot{\mathbf{r}}^i - \Omega_{i/e}^i \dot{\mathbf{r}}^i) - C_i^e \Omega_{i/e}^i (\dot{\mathbf{r}}^i - \Omega_{i/e}^i \mathbf{r}^i) \\ &= C_i^e (\ddot{\mathbf{r}}^i - 2\Omega_{i/e}^i \dot{\mathbf{r}}^i + \Omega_{i/e}^i \Omega_{i/e}^i \mathbf{r}^i) \end{aligned} \quad (5.7)$$

where, in the second line, the Earth's rotation rate is assumed to be constant

$$\dot{\Omega}_{i/e}^i = 0 \quad (5.8)$$

The time rate of change of the direction cosine matrix  $C_e^n$  is

$$\dot{C}_e^n = -\Omega_{e/n}^n C_e^n \quad (5.9)$$

Substituting Eqs. (5.6), (5.7), and (5.9) into Eq. (5.2) and using Eq. (5.1) yield, after some manipulations,

$$\begin{aligned} \dot{\mathbf{v}}^n &= C_e^n C_i^e (\ddot{\mathbf{r}}^i - 2\Omega_{i/e}^i \dot{\mathbf{r}}^i + \Omega_{i/e}^i \Omega_{i/e}^i \mathbf{r}^i) - \Omega_{e/n}^n C_e^n C_n^e \mathbf{v}^n \\ &= C_i^n (\ddot{\mathbf{r}}^i - 2\Omega_{i/e}^i C_n^i \mathbf{v}^n - 2\Omega_{i/e}^i \Omega_{i/e}^i \dot{\mathbf{r}}^i + \Omega_{i/e}^i \Omega_{i/e}^i \mathbf{r}^i) - \Omega_{e/n}^n \mathbf{v}^n \\ &= C_i^n (\ddot{\mathbf{r}}^i - 2\Omega_{i/e}^i C_n^i \mathbf{v}^n - \Omega_{i/e}^i \Omega_{i/e}^i \mathbf{r}^i) - C_i^n \Omega_{e/n}^i C_n^i \mathbf{v}^n \\ &= C_i^n \left[ \ddot{\mathbf{r}}^i - (\Omega_{e/n}^i + 2\Omega_{i/e}^i) C_n^i \mathbf{v}^n - \Omega_{i/e}^i \Omega_{i/e}^i \mathbf{r}^i \right] \end{aligned} \quad (5.10)$$

where, in the second step in this equation, the following relationship is used:

$$\dot{\mathbf{r}}^i = \Omega_{i/e}^i \mathbf{r}^i + C_n^i \mathbf{v}^n$$

This equation can be obtained using Eqs. (5.1) and (5.4).

The specific force  $\mathbf{f}^n$  is the sensed output of accelerometers transformed into the navigation frame. This specific force is a combination of inertial and gravitational acceleration:

$$\mathbf{f}^n = C_i^n \ddot{\mathbf{r}}^i - \mathbf{G}^n \quad (5.11)$$

Gravitational acceleration can be expressed as the sum of gravity and centripetal acceleration (induced by the Earth's rotation):

$$\mathbf{G}^n = \mathbf{g}^n + \Omega_{i/e}^n \Omega_{i/e}^n \mathbf{r}^n \quad (5.12)$$

Using the similarity transform on the product of Earth rates

$$\Omega_{i/n}^n \Omega_{i/n}^n = C_i^n \Omega_{i/n}^i C_n^i C_i^n \Omega_{i/n}^i C_n^i \quad (5.13)$$

and combining Eqs. (5.11) and (5.12) and rearranging terms yield

$$\begin{aligned} \ddot{\mathbf{r}}^i &= C_n^i (\mathbf{f}^n + \mathbf{G}^n) \\ &= C_n^i (\mathbf{f}^n + \mathbf{g}^n + C_i^n \Omega_{i/e}^i \Omega_{i/e}^i C_n^i \mathbf{r}^n) \\ &= C_n^i (\mathbf{f}^n + \mathbf{g}^n) + \Omega_{i/e}^i \Omega_{i/e}^i \mathbf{r}^i \end{aligned} \quad (5.14)$$

Substituting Eq. (5.14) into Eq. (5.10) yields

$$\begin{aligned}\dot{\mathbf{v}}^n &= C_i^n | C_n^i (\mathbf{f}^n + \mathbf{g}^n) - (\Omega_{e/n}^i + 2\Omega_{i/e}^i) C_n^i \mathbf{v}^n \\ &= \mathbf{f}^n + \mathbf{g}^n - C_i^n (\Omega_{e/n}^i + 2\Omega_{i/e}^i) C_n^i \mathbf{v}^n\end{aligned}$$

which, after using the similarity transformation for the angular rates, becomes

$$\dot{\mathbf{v}}^n = \mathbf{f}^n - (\Omega_{e/n}^n + 2\Omega_{i/e}^n) \mathbf{v}^n + \mathbf{g}^n \quad (5.15)$$

This equation provides the basic description of the Earth relative velocity evolution in a local level navigation frame, and is applicable to both local-level geodetic and to wander-azimuth frames. Specification of the frame is only necessary when writing out components and defining the  $\Omega$  rotation matrix elements, thus, this equation is also applicable to either gimballed or strap-down navigation system mechanizations.

### 5.2.2 Attitude Equations

In this treatment of integrated navigation systems, the attitude dynamics equation is represented by the direction cosine matrix differential equation. The general form for the direction cosine matrix rate was described in Sec. 2.4. In this form, once initialized, it can be integrated without specific expressions for each of its elements.

This attitude dynamics evolution is obtained from the following:

$$\dot{C}_b^n = -\Omega_{b/n}^n C_b^n \quad (5.16)$$

The skew-symmetric matrix  $\Omega_{b/n}^n$  is formed from the rotation vector  $\boldsymbol{\omega}_{b/n}^n$ . This vector is obtained using the angular velocity addition theorem

$$\boldsymbol{\omega}_{b/n}^n = \boldsymbol{\omega}_{i/n}^n - C_b^n \boldsymbol{\omega}_{i/b}^b \quad (5.17)$$

The rotation vector  $\boldsymbol{\omega}_{i/b}^b$  represents the outputs of the strap-down gyros.

Several methods can be used to evolve attitudes. These include integration of 1) six elements of Eq. (5.16) with the remaining three computed as in Problem 2.5; 2) integration of four quaternions or other equivalent representations for the direction cosine matrix transformation (see Sec. 2.5); or 3) integration of three Euler-angle equations (see Problem 5.6). Which method to select is a tradeoff involving considerations of vehicle operating environment, computer capability, and accuracy requirements.

### 5.2.3 Position Equations

The position evolution is expressed as the following direction cosine matrix differential equation [Eq. (5.9)]:

$$\dot{C}_e^n = -\Omega_{e/n}^n C_e^n$$

The elements of this matrix were given in Chapter 3 for different definitions of the  $C_e^n$  matrix. As with the  $C_b^n$  matrix, several methods are available to evolve positions.

### 5.3 Perturbation Form of Navigation System Error Equations

In this section, the nonlinear navigation state equations for velocity, attitude, and position are linearized to obtain linear error model forms. The linearization approach used in this section is to develop linear equations that represent perturbations about a nominal solution as developed earlier in Chapter 2. The resulting equations are further specialized to wander-azimuth frames, with the vertical  $z$  axis directed up, as presented in Chapter 3.

#### 5.3.1 Velocity Error Equations

The navigation frame velocity equation (5.15) is rewritten using the vector cross-product form for the rotation rates as

$$\dot{\mathbf{v}}^n = \mathbf{f}^n - (\boldsymbol{\omega}_{e/n}^n + 2\boldsymbol{\omega}_{i/e}^n) \times \mathbf{v}^n + \mathbf{g}^n$$

The computed velocity is maintained within the navigation system and is implemented as the same equation. As a result of initialization, inertial sensor, and other errors, this computed velocity is imperfect—corrupted. This corruption is represented with overbars over the variables in this equation as

$$\bar{\dot{\mathbf{v}}}^n = \bar{\mathbf{f}}^n - (\bar{\boldsymbol{\omega}}_{e/n}^n + 2\bar{\boldsymbol{\omega}}_{i/e}^n) \times \bar{\mathbf{v}}^n + \bar{\mathbf{g}}^n \quad (5.18)$$

The computed velocity is defined for the perturbation approach as the true value plus a linear perturbation velocity error  $\delta\mathbf{v}^n$  in the navigation frame as

$$\bar{\mathbf{v}}^n \equiv \mathbf{v}^n + \delta\mathbf{v}^n \quad (5.19)$$

Likewise, for the other terms in Eq. (5.18), the following are expressions for their computed values:

$$\bar{\mathbf{f}}^n = \bar{C}_b^n \bar{\mathbf{f}}^b = [I - (\boldsymbol{\phi} \times)] C_b^n (\mathbf{f}^b + \delta\mathbf{f}^b) \approx [I - (\boldsymbol{\phi} \times)] \mathbf{f}^n + \delta\mathbf{f}^n \quad (5.20)$$

$$\bar{\boldsymbol{\omega}}_{e/n}^n = \boldsymbol{\omega}_{e/n}^n + \delta\boldsymbol{\omega}_{e/n}^n \quad (5.21)$$

$$\bar{\boldsymbol{\omega}}_{i/e}^n = \boldsymbol{\omega}_{i/e}^n + \delta\boldsymbol{\omega}_{i/e}^n \quad (5.22)$$

$$\bar{\mathbf{g}}^n = \mathbf{g}^n + \delta\mathbf{g}^n \quad (5.23)$$

Substituting these equations into the corrupted form in Eq. (5.18), subtracting the uncorrupted equation from this result, and eliminating products of error quantities yield

$$\delta\dot{\mathbf{v}}^n = -(\delta\boldsymbol{\omega}_{e/n}^n + 2\delta\boldsymbol{\omega}_{i/e}^n) \times \mathbf{v}^n - (\boldsymbol{\omega}_{e/n}^n + 2\boldsymbol{\omega}_{i/e}^n) \times \delta\mathbf{v}^n + \mathbf{f}^n \times \boldsymbol{\phi} + \delta\mathbf{f}^n + \delta\mathbf{g}^n \quad (5.24)$$

The error rotation vectors  $\delta\boldsymbol{\omega}_{e/n}^n$  and  $\delta\boldsymbol{\omega}_{i/e}^n$  are functions of velocity and position perturbations, respectively, and are required to complete the expression for the velocity error equation. The result in Eq. (5.24) is in general applicable

to any three-axis navigation system; however, at this point it is specialized to a wander-azimuth referenced system with the  $z$  axis up.

In formulating the rotation vector  $\omega_{e/n}^n$  for the purposes of linearization, the Earth radii are approximated by a nominal radius  $R$

$$\omega_{e/n}^n \approx \begin{bmatrix} -\frac{v_y^n}{R} \\ \frac{v_x^n}{R} \\ \frac{v_z^n}{R} \end{bmatrix} \quad (5.25)$$

The variation of this equation with respect to velocity and Earth radius yields

$$\delta\omega_{e/n}^n \approx \begin{bmatrix} -\frac{\delta v_y^n}{R} - \frac{\rho_x}{R} \delta h \\ \frac{\delta v_x^n}{R} - \frac{\rho_y}{R} \delta h \\ \delta \rho_z \end{bmatrix} \quad (5.26)$$

where  $\delta R \approx \delta h$  and the notation  $\omega_{e/m}^n \equiv \boldsymbol{\rho}$  has been retained from Example 3.2. The last element in Eq. (5.25) represents the vehicle transport velocity about the vertical axis. It was shown in Chapter 3 that, for wander-azimuth mechanizations, this term is specified as zero; however, for geodetic mechanizations, this is not true.

The error in the Earth rotation vector  $\omega_{i/e}^n \equiv \boldsymbol{\Omega}$  is expressed in vector cross-product form (see Problem 4.8).

$$\delta\omega_{i/e}^n = \omega_{i/e}^n \times \delta\boldsymbol{\theta} \quad (5.27)$$

Substituting Eqs. (5.25–5.27) into Eq. (5.24), the perturbation velocity error equation form for wander-azimuth mechanizations is obtained.

### 5.3.2 Attitude Error Equations

Attitude error equations are derived as the  $\phi$  or tilt error formulation. The computed, or corrupted, body-to-navigation frame transformation matrix is represented as (see Problem 3.4)

$$\bar{C}_b^n \equiv [I - (\boldsymbol{\phi} \times)] C_b^n \quad (5.28)$$

The error in this matrix is

$$\begin{aligned} \delta C_b^n &= \bar{C}_b^n - C_b^n \\ &= -(\boldsymbol{\phi} \times) C_b^n \end{aligned} \quad (5.29)$$

The dynamic equation for  $\phi$  is obtained by taking the derivative of both lines of this equation, then equating the result and solving for  $\dot{\phi}$  from the second row

expression in Eq. (5.29):

$$\delta \dot{C}_b^n = -(\dot{\phi} \times) C_b^n - (\phi \times) \dot{C}_b^n \quad (5.30)$$

The derivative of the body-to-navigation frame transformation matrix was given in Eq. (5.16) as

$$\dot{C}_b^n = -\Omega_{b/n}^n C_b^n$$

Substituting this equation into Eq. (5.30) results in

$$\delta \dot{C}_b^n = -[(\dot{\phi} \times) - (\phi \times) \Omega_{b/n}^n] C_b^n \quad (5.31)$$

From the first row and using the corresponding matrix derivative equation, as in Eq. (5.16), yields

$$\begin{aligned} \delta \dot{C}_b^n &= \dot{\bar{C}}_b^n - \dot{C}_b^n \\ &= -\bar{\Omega}_{b/n}^n \bar{C}_b^n + \Omega_{b/n}^n C_b^n \\ &= -\bar{\Omega}_{b/n}^n [I - (\phi \times)] C_b^n + \Omega_{b/n}^n C_b^n \\ &= -[\bar{\Omega}_{b/n}^n - \Omega_{b/n}^n - \bar{\Omega}_{b/n}^n (\phi \times)] C_b^n \\ &\approx -[\bar{\Omega}_{b/n}^n - \Omega_{b/n}^n - \Omega_{b/n}^n (\phi \times)] C_b^n \end{aligned} \quad (5.32)$$

Equating Eq. (5.31) and this equation yields

$$(\dot{\phi} \times) - (\phi \times) \Omega_{b/n}^n = \bar{\Omega}_{b/n}^n - \Omega_{b/n}^n - \Omega_{b/n}^n (\phi \times)$$

or

$$(\dot{\phi} \times) = -\Omega_{b/n}^n (\phi \times) + (\phi \times) \Omega_{b/n}^n + (\bar{\Omega}_{b/n}^n - \Omega_{b/n}^n) \quad (5.33)$$

Equivalently, in vector form,

$$\dot{\phi} = \phi \times \omega_{b/n}^n + (\bar{\omega}_{b/n}^n - \omega_{b/n}^n) \quad (5.34)$$

Expanding angular rates using the angular velocity addition theorem to account for gyro error, the first of the terms in parentheses in Eq. (5.34) is expressed as

$$\begin{aligned} \bar{\omega}_{b/n}^n &= \bar{\omega}_{i/n}^n - \bar{C}_b^n \bar{\omega}_{i/b}^b \\ &\equiv \bar{\omega}_{i/n}^n - \bar{C}_b^n \omega_{i/b}^b + \epsilon^n \end{aligned} \quad (5.35)$$

The second term in Eq. (5.34) is the same rate without error or corruption:

$$\bar{\boldsymbol{\omega}}_{b/n}^n = \boldsymbol{\omega}_{i/n}^n - C_b^n \boldsymbol{\omega}_{i/b}^b \quad (5.36)$$

Subtracting this equation from Eq. (5.35) yields

$$\begin{aligned} \bar{\boldsymbol{\omega}}_{b/n}^n - \boldsymbol{\omega}_{b/n}^n &= \bar{\boldsymbol{\omega}}_{i/n}^n - \boldsymbol{\omega}_{i/n}^n - [I - (\boldsymbol{\phi} \times)] C_b^n \boldsymbol{\omega}_{i/b}^b + \boldsymbol{\varepsilon}^n + C_b^n \boldsymbol{\omega}_{i/b}^b \\ &\equiv \bar{\boldsymbol{\omega}}_{i/n}^n - \boldsymbol{\omega}_{i/n}^n + (\boldsymbol{\phi} \times) C_b^n \boldsymbol{\omega}_{i/b}^b + \boldsymbol{\varepsilon}^n \end{aligned} \quad (5.37)$$

Define, temporarily, a rotation rate error vector as

$$\delta\boldsymbol{\omega} = \bar{\boldsymbol{\omega}}_{i/n}^n - \boldsymbol{\omega}_{i/n}^n \quad (5.38)$$

Then, Eq. (5.37) can be rewritten as

$$\bar{\boldsymbol{\omega}}_{b/n}^n - \boldsymbol{\omega}_{b/n}^n = \delta\boldsymbol{\omega} + (\boldsymbol{\phi} \times) C_b^n \boldsymbol{\omega}_{i/b}^b + \boldsymbol{\varepsilon}^n \quad (5.39)$$

or, in vector form,

$$\bar{\boldsymbol{\omega}}_{b/n}^n - \boldsymbol{\omega}_{b/n}^n = \delta\boldsymbol{\omega} + \boldsymbol{\phi} \times C_b^n \boldsymbol{\omega}_{i/b}^b + \boldsymbol{\varepsilon}^n \quad (5.40)$$

Substituting this equation into Eq. (5.34) yields

$$\begin{aligned} \dot{\boldsymbol{\phi}} &= \boldsymbol{\phi} \times \boldsymbol{\omega}_{b/n}^n + \delta\boldsymbol{\omega} + \boldsymbol{\phi} \times C_b^n \boldsymbol{\omega}_{i/b}^b + \boldsymbol{\varepsilon}^n \\ &= \delta\boldsymbol{\omega} + \boldsymbol{\phi} \times (\boldsymbol{\omega}_{b/n}^n + \boldsymbol{\omega}_{b/n}^n) + \boldsymbol{\varepsilon}^n \end{aligned} \quad (5.41)$$

Recognizing that, from the angular velocity addition theorem,

$$\boldsymbol{\omega}_{b/n}^n + C_b^n \boldsymbol{\omega}_{i/b}^b = \boldsymbol{\omega}_{b/n}^n + \boldsymbol{\omega}_{i/b}^b = \boldsymbol{\omega}_{i/n}^n \quad (5.42)$$

then, Eq. (5.41) becomes

$$\dot{\boldsymbol{\phi}} = \delta\boldsymbol{\omega} + \boldsymbol{\phi} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\varepsilon}^n \quad (5.43)$$

It remains to determine the rotation error vector  $\delta\boldsymbol{\omega}$ . From its definition in Eq. (5.38),

$$\begin{aligned} \delta\boldsymbol{\omega} &= (\bar{\boldsymbol{\omega}}_{e/n}^n + \bar{\boldsymbol{\omega}}_{i/e}^n) - (\boldsymbol{\omega}_{e/n}^n + \boldsymbol{\omega}_{i/e}^n) \\ &= (\bar{\boldsymbol{\omega}}_{e/n}^n - \boldsymbol{\omega}_{e/n}^n) - (\bar{\boldsymbol{\omega}}_{i/e}^n - \boldsymbol{\omega}_{i/e}^n) \\ &= \delta\boldsymbol{\rho} + \boldsymbol{\omega}_{i/e}^n \times \delta\boldsymbol{\theta} \end{aligned} \quad (5.44)$$

Substituting this equation into Eq. (5.43) yields the following attitude error equation:

$$\dot{\boldsymbol{\phi}} = \delta\boldsymbol{\rho} + \boldsymbol{\omega}_{i/e}^n \times \delta\boldsymbol{\theta} + \boldsymbol{\phi} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\varepsilon}^n \quad (5.45)$$

In obtaining this equation for the tilt error, assumptions used to obtain the velocity error equation were not required; thus, it is generally applicable to any three-axis navigation system error model.

### 5.3.3 Position Error Equations

Position errors are obtained from the Earth-to-navigation frame direction cosine matrix. The computed or corrupted direction cosine matrix is represented as

$$\bar{C}_e^n \equiv [I - (\delta\boldsymbol{\theta} \times)]C_e^n \quad (5.46)$$

or the error in this matrix is

$$\begin{aligned}\delta C_e^n &= \bar{C}_e^n - C_e^n \\ &= -(\delta\boldsymbol{\theta} \times)C_e^n\end{aligned} \quad (5.47)$$

The dynamic equation for  $\delta\theta$  is obtained in a similar fashion to that used for the tilt equation above.

Taking the time derivative of the second row of this equation yields

$$\begin{aligned}\delta \dot{C}_e^n &= -(\delta\dot{\boldsymbol{\theta}} \times)C_e^n - (\delta\boldsymbol{\theta} \times)\dot{C}_e^n \\ &= -[(\delta\dot{\boldsymbol{\theta}} \times) - (\delta\boldsymbol{\theta} \times)\Omega_{e/n}^n]C_e^n\end{aligned} \quad (5.48)$$

Returning to Eq. (5.47) and taking the time derivative of the first row and using the corresponding definition for the derivative of a direction cosine matrix yields

$$\begin{aligned}\delta \dot{C}_e^n &= \dot{\bar{C}}_e^n - \dot{C}_e^n \\ &= -\bar{\Omega}_{e/n}^n \bar{C}_e^n + \Omega_{e/n}^n C_e^n \\ &= -\bar{\Omega}_{e/n}^n [I - (\delta\boldsymbol{\theta} \times)]C_e^n + \Omega_{e/n}^n C_e^n \\ &= -[\bar{\Omega}_{e/n}^n - \Omega_{e/n}^n - \bar{\Omega}_{e/n}^n (\delta\boldsymbol{\theta} \times)]C_e^n \\ &\approx -[\bar{\Omega}_{e/n}^n - \Omega_{e/n}^n - \Omega_{e/n}^n (\delta\boldsymbol{\theta} \times)]C_e^n\end{aligned} \quad (5.49)$$

Equating Eq. (5.48) and this equation,

$$(\delta\dot{\boldsymbol{\theta}} \times) - (\delta\boldsymbol{\theta} \times)\Omega_{e/n}^n = \bar{\Omega}_{e/n}^n - \Omega_{e/n}^n - \Omega_{e/n}^n (\delta\boldsymbol{\theta} \times)$$

or

$$(\delta\dot{\boldsymbol{\theta}} \times) = (\delta\boldsymbol{\rho} \times) - \Omega_{e/n}^n (\delta\boldsymbol{\theta} \times) + (\delta\boldsymbol{\theta} \times) \Omega_{e/n}^n \quad (5.50)$$

Equivalently, in vector form,

$$\delta\dot{\boldsymbol{\theta}} = \delta\boldsymbol{\rho} - \boldsymbol{\omega}_{e/n}^n \times \delta\boldsymbol{\theta} \quad (5.51)$$

As with the tilt error equation, assumptions used to obtain the velocity error equation were not required; thus, this angular position error dynamic equation is generally applicable to any error model with three angular errors. However,

this form is not generally applicable to angular position error with only two angular errors, that is, latitude and longitude errors (see Exercise 5.14).

### 5.3.4 Perturbation Error Summary

In the following, it will be assumed that the variation of the transport rate is zero:

$$\delta\rho_z \equiv 0 \quad (5.52)$$

thus specializing the following to the perturbation form for velocity error representation given in Eq. (5.19) and wander-azimuth mechanizations, where the vertical transport rate  $\rho_z$  is defined as zero. This assumption is necessary to obtain a closed set of equations, that is, as many equations as unknown error variables. Figure 5.2 presents a summary of this perturbation form of navigation system error equations and is found in the literature [6].

The assumption  $\delta\rho_z \equiv 0$  results in an error model with four attitude error states. This is an additional error state compared to a form based on an alternate velocity error representation presented later in this chapter. This additional state requires more processing time and storage for its implementation in a Kalman filter algorithm (see Chapter 8 problems). Whether or not to implement a model of this form requires consideration of the impact of its implementation on computer capability.

### Example 5.1 Approximations, Navigation Error, and Schuler Period

Consider a nonaccelerating case such that the following approximations hold:

$$\delta\dot{v}_y = g\phi_x + \delta f_y$$

$$\dot{\phi}_x = -\frac{1}{R}\delta v_y + \varepsilon_x$$

These equations are decoupled from  $x$  velocity component equations and can be examined separately. Reducing these equations to a single second-order differential equation, assuming that accelerometer and gyro errors are constant, yields

$$\begin{aligned} \delta\ddot{v}_y &= g\dot{\phi}_x \\ &= -\frac{g}{R}\delta v_y + g\varepsilon_x \end{aligned}$$

or

$$\delta\ddot{v}_y + \frac{g}{R}\delta v_y = g\varepsilon_x$$

Assuming a solution of the form

$$\delta v_y = A \sin \omega t + B \cos \omega t + C$$

$$\begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & -\rho_y \\
 0 & 0 & 0 & 0 & 0 & 0 & \rho_x \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 \frac{\delta\theta_x}{\delta\theta_y} & -2(\dot{\psi}_y\Omega_x + \dot{\nu}_x\Omega_z) & 2\nu_y\Omega_x & -\frac{\nu_x\nu_z}{R^2} & 2\Omega_z & -(\rho + 2\Omega)_y & 2\nu_z\Omega_x \\
 \frac{\delta\theta_y}{\delta\theta_z} & 2\nu_x\Omega_y & -2(\dot{\psi}_z\Omega_x + \dot{\nu}_z\Omega_y) & -\frac{\nu_y\nu_z}{R^2} & -2\Omega_z & (\rho + 2\Omega)_x & 2\nu_z\Omega_y \\
 \frac{d}{dt}\frac{\delta\theta_x}{\delta\theta_z} & 2\nu_x\Omega_z & 2\nu_y\Omega_z & \frac{\nu_x^2 + \nu_y^2 + 2g}{R^2} & 2(\rho + \Omega)_z & 0 & -f_z \\
 \phi_x & 0 & -\Omega_z & \frac{\nu_y}{R^2} & 0 & -\frac{1}{R} & f_x \\
 \phi_y & \Omega_z & 0 & -\frac{\nu_z}{R^2} & 1 & 0 & -f_y \\
 \phi_z & -\Omega_y & -\Omega_y & 0 & 0 & 0 & f_z \\
 \rho_y & \rho_y & 0 & 0 & 0 & 0 & 0
 \end{bmatrix} = \begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{R} \\
 0 & 0 & 0 & 0 & 0 & 0 & \frac{\nu_x}{R} \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 \frac{\delta\theta_x}{\delta\theta_y} & \frac{\delta f_x}{\delta\theta_x} & \frac{\delta g_x}{\delta\theta_x} & 0 & 0 & 0 & \frac{\rho_x}{\delta\theta_y} \\
 \frac{\delta\theta_y}{\delta\theta_z} & \frac{\delta f_y}{\delta\theta_y} + \frac{\delta g_y}{\delta\theta_z} & \frac{\delta g_z}{\delta\theta_z} & 0 & 0 & \varepsilon_x & 0 \\
 \frac{d}{dt}\frac{\delta\theta_x}{\delta\theta_z} & 0 & 0 & 0 & 0 & \varepsilon_y & \varepsilon_z \\
 \phi_x & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix} + \begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & -2(\dot{\psi}_x\Omega_x + \dot{\nu}_y\Omega_y) \\
 0 & 0 & 0 & 0 & 0 & 0 & \phi_x \\
 0 & 0 & 0 & 0 & 0 & 0 & \phi_y \\
 0 & 0 & 0 & 0 & 0 & 0 & \phi_z \\
 0 & 0 & 0 & 0 & 0 & 0 & -\Omega_z
 \end{bmatrix}$$

Fig. 5.2 Perturbation form of navigation system error equations ( $\delta\theta_2 = 0$ ).

yields

$$\delta v_y = \delta v_y(0) \cos \omega t + [g\phi_x(0) + \delta f_x] \frac{\sin \omega t}{\omega} + g\varepsilon_x \frac{(1 - \cos \omega t)}{\omega^2}$$

From this equation, it is seen that the initial tilt error  $\phi_x(0)$  and the  $y$  component of accelerometer bias  $\delta f_y$  contribute as a sum to the velocity error.

Returning to the  $\delta\dot{v}_y$  equation, velocity errors are governed by the frequency  $\omega$ . Using nominal values for gravity and Earth radius, this frequency is approximated by

$$\omega_s = \sqrt{\frac{g}{R}} \approx \sqrt{\frac{9.798}{6,378,000}} \approx \frac{2\pi \text{ rad}}{84(60) \text{ s}}$$

This 84-min characteristic frequency is termed the Schuler period.

## 5.4 Navigation System Attitude Error Equations: $\psi$ Formulation

In Sec. 5.3, mechanization equations were linearized using the perturbation approach. This approach produced error dynamic equations in the  $\phi$  formulation. Another useful form is the  $\psi$  formulation, which yields simpler attitude error dynamic equations and offers the potential for reduced computational demands of the onboard computer.

The  $\psi$  formulation defines attitude error as

$$\boldsymbol{\psi} \equiv \boldsymbol{\phi} - \boldsymbol{\delta\theta} \quad (5.53)$$

This error angle vector represents the combination or local-level tilt angular error and an angular position error relative to Earth center.

The error dynamic equation can be obtained by taking the derivative of the preceding equation, which yields

$$\dot{\boldsymbol{\psi}} = \dot{\boldsymbol{\phi}} - \dot{\boldsymbol{\delta\theta}} \quad (5.54)$$

Substituting results from Eqs. (5.46) and (5.52) into this equation yields

$$\begin{aligned} \dot{\boldsymbol{\psi}} &= \boldsymbol{\delta\rho} + \boldsymbol{\omega}_{i/e}^n \times \boldsymbol{\delta\theta} + \boldsymbol{\phi} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\epsilon}^n - \boldsymbol{\delta\rho} + \boldsymbol{\omega}_{e/n}^n \times \boldsymbol{\delta\theta} \\ &= (\boldsymbol{\omega}_{i/e}^n + \boldsymbol{\omega}_{e/n}^n) \times \boldsymbol{\delta\theta} + \boldsymbol{\phi} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\epsilon}^n \\ &= -\boldsymbol{\delta\theta} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\phi} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\epsilon}^n \\ &= (\boldsymbol{\phi} - \boldsymbol{\delta\theta}) \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\epsilon}^n \end{aligned}$$

or

$$\dot{\boldsymbol{\psi}} = \boldsymbol{\psi} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\epsilon}^n \quad (5.55)$$

This  $\psi$  formulation requires fewer terms than the tilt error equation in Eq. (5.45) to form attitude error dynamics. This equation is generally applicable for any three-axis navigation frame mechanization.

## 5.5 Navigation System Error Equations Using Alternative Velocity Error

In this section, navigation system equations developed in Sec. 5.1 are linearized using an alternative form for velocity error. The velocity error model that results from using this alternate velocity error representation, combined with the results of the preceding section, produces an error model that contains fewer elements and thus is a candidate for easier implementation. A summary of the resulting navigation system error equations is presented.

### 5.5.1 Alternative Velocity Error

The velocity given in Eq. (5.15) is again assumed for true and computed velocities. The computed velocity in this case is represented as

$$\bar{v}^n \equiv [I - (\delta\theta \times)]v^n + \delta v^1 \quad (5.56)$$

Note, the superscript 1 in this equation is used to distinguish this velocity error from that used in Eq. (5.19). The computed values for specific force, rates, and gravity, expressed in terms of true values and perturbation errors, remain unchanged from those given in Eqs. (5.20–5.23). Substituting this equation and Eqs. (5.20–5.23) into Eq. (5.18) and then subtracting the uncorrupted form of this equation from the result yield the following equation for the velocity error:

$$\begin{aligned} \delta\dot{v}^1 - \delta\dot{\theta} \times v^n - \delta\theta \times \dot{v}^n &= -\phi \times f^n + \delta f^n - (\omega_{e/n}^n + \omega_{i/e}^n) \times (\delta v^1 - \delta\theta \times v^n) \\ &\quad - (\delta\omega_{e/n}^n + 2\delta\omega_{i/e}^n) \times v^n + \delta g^n \end{aligned} \quad (5.57)$$

This equation is simplified by using Eq. (5.53) and the velocity equation given in Eq. (5.15). Substituting these equations into this equation yields

$$\begin{aligned} \delta\dot{v}^1 - (\delta\omega_{e/n}^n - \omega_{e/n}^n \times \delta\theta) \times v^n - \delta\theta \times [f^n - (\omega_{e/n}^n + 2\omega_{i/e}^n) \times v^n + g^n] \\ = -\phi \times f^n + \delta f^n - (\omega_{e/n}^n + \omega_{i/e}^n) \times (\delta v^1 - \delta\theta \times v^n) \\ - (\delta\omega_{e/n}^n + 2\delta\omega_{i/e}^n) \times v^n + \delta g^n \end{aligned} \quad (5.58)$$

Cancelling terms and using Eq. (5.27) with the following vector identities

$$(\omega_{e/n}^n \times \delta\theta) \times v^n + \delta\theta \times (\omega_{e/n}^n \times v^n) = \omega_{e/n}^n \times (\delta\theta \times v^n) \quad (5.59)$$

$$\delta\theta \times (\omega_{i/e}^n \times v^n) = \omega_{i/e}^n \times (\delta\theta \times v^n) - (\omega_{i/e}^n \times \delta\theta) \times v^n \quad (5.60)$$

gives the following form of the velocity error equation:

$$\begin{aligned}\delta\dot{v}^1 = & -(\delta\boldsymbol{\theta} - \boldsymbol{\phi}) \times \mathbf{f}^n + \delta\mathbf{f}^n - (\boldsymbol{\omega}_{e/n}^n + 2\boldsymbol{\omega}_{i/e}^n) \times \delta\mathbf{v}^1 \\ & + \delta\mathbf{g}^n + \delta\boldsymbol{\theta} \times \mathbf{g}^n\end{aligned}\quad (5.61)$$

Using the definition of the  $\psi$  attitude error in Eq. (5.53) and regrouping terms, this equation becomes

$$\delta\dot{v}^1 = -\mathbf{g}^n \times \delta\boldsymbol{\theta} - (\boldsymbol{\omega}_{e/n}^n + 2\boldsymbol{\omega}_{i/e}^n) \times \delta\mathbf{v}^1 + \mathbf{f}^n \times \boldsymbol{\psi} + \delta\mathbf{f}^n + \delta\mathbf{g}^n \quad (5.62)$$

By momentarily redefining the gravity error as

$$\delta\mathbf{g}^1 \equiv \delta\mathbf{g}^n - \mathbf{g} \times \delta\boldsymbol{\theta} \quad (5.63)$$

Eq. (5.62) can be written as

$$\delta\dot{v}^1 = -(\boldsymbol{\omega}_{e/n}^n + 2\boldsymbol{\omega}_{i/e}^n) \times \delta\mathbf{v}^1 + \mathbf{f}^n \times \boldsymbol{\psi} + \delta\mathbf{f}^n + \delta\mathbf{g}^1 \quad (5.64)$$

Comparing this equation to Eq. (5.24) suggests the rotation rates  $\boldsymbol{\omega}_{e/n}^n$  and  $\boldsymbol{\omega}_{i/e}^n$  are known and without error ( $\delta\boldsymbol{\omega}_{e/n}^n$  and  $\delta\boldsymbol{\omega}_{i/e}^n$  are zero), and the tilt error vector  $\boldsymbol{\phi}$  is replaced by the  $\boldsymbol{\psi}$  form for attitude error.

### 5.5.2 Attitude Error for Alternate Velocity Error

The attitude error is given in Eq. (5.55) for the  $\boldsymbol{\psi}$  form.

### 5.5.3 Position Error for Alternate Velocity Error

The angular position error is given in Eq. (5.51). Using the velocity error representation given in Eq. (5.56), position error dynamics become

$$\begin{aligned}\delta\dot{\theta} = & \begin{bmatrix} -(\delta v_y^1 + v_z \delta\theta_x - v_x \delta\theta_z)/R + (v_y/R^2)\delta h \\ (\delta v_x^1 - v_z \delta\theta_y + v_y \delta\theta_z)/R - (v_x/R^2)\delta h \\ \delta\rho_z \end{bmatrix} \\ & + \begin{bmatrix} 0 & \rho_z & -\rho_y \\ -\rho_z & 0 & \rho_x \\ \rho_y & -\rho_x & 0 \end{bmatrix} \begin{bmatrix} \delta\theta_x \\ \delta\theta_y \\ \delta\theta_z \end{bmatrix}\end{aligned}\quad (5.65)$$

Assuming a wander-azimuth frame implementation with the vertical transport rate  $\rho_z$  being zero and using Eq. (5.25) for transport rates as functions of velocity

components, the horizontal components of the position error equations become

$$\delta\dot{\theta}_x = -\frac{v_z}{R} \delta\theta_x + \frac{v_y}{R^2} \delta h - \frac{1}{R} \delta v_y^1 \quad (5.66)$$

$$\delta\dot{\theta}_y = -\frac{v_z}{R} \delta\theta_y - \frac{v_x}{R^2} \delta h - \frac{1}{R} \delta v_x^1 \quad (5.67)$$

The altitude rate error is same as the vertical velocity error

$$\delta\dot{h} = \delta v_z^n \quad (5.68)$$

which, from Eq. (5.56), allows the vertical component of position error to be expressed as

$$\delta\dot{h} = -v_y \delta\theta_x + v_x \delta\theta_y + \delta v_z^1 \quad (5.69)$$

#### 5.5.4 Error Equations for Alternate Velocity Error Summary

Figure 5.3 presents a summary of navigation system error equations using the alternate velocity error representation given in Eq. (5.56). The vertical transport rate  $\rho_z$  is assumed to be zero. At this point in the development of these equations, no assumptions concerning  $\delta\theta_z$ ,  $\delta\rho_z$ , etc. have been made.

This error model, using the alternate velocity error representation, appears in the literature [7]. A similar formulation, using the  $\phi$  equivalent attitude error representation instead of the  $\psi$  form presented here, is used for a case study on transfer alignment presented in Chapter 13. The model summarized in Fig. 5.3 requires one less error state than that presented in Fig. 5.2 which required four attitude error states. Therefore, if computational constraints require an implementation with fewer states, this model might warrant further consideration.

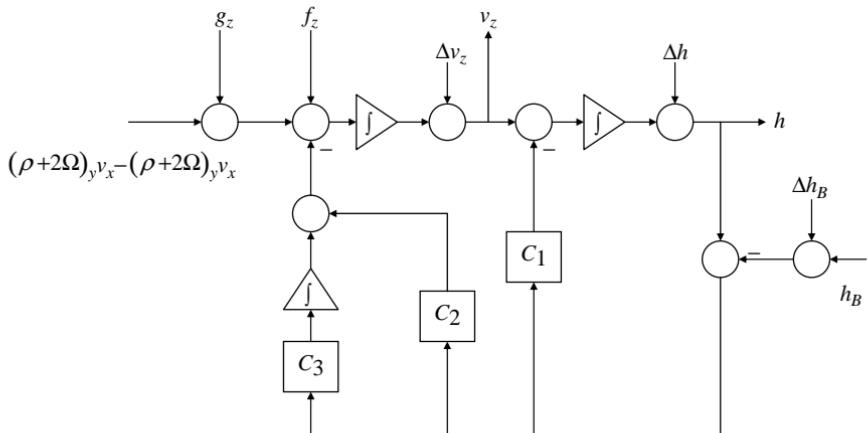
## 5.6 Vertical Channel

Current inertial navigation systems utilize barometric pressure to stabilize and control the vertical channel—altitude, but inaccurate vertical velocity, etc. Barometric pressure is considered to be relatively stable, but inaccurate measure of absolute altitude above sea level. A feedback control system is implemented that uses proportional/derivative and integral feedback of the difference between system and barometric altitudes. The integral feedback eliminates steady-state error between system and barometric altitudes.

A representative vertical channel mechanization is illustrated in Fig. 5.4. This mechanization is simplified to include those elements common to the various mechanizations available. Differences between this and other mechanizations that exist address different approaches to dynamically adjusting the feedback coefficients  $C_i$ . During rapid altitude changes, the barometric altitude might be in error, and the coefficients are adjusted to rely more on the pure inertial solution for altitude. The delta values in the figure, that is,  $\Delta h_B$ , indicate corrections that are added to adjust the loop's values for the vertical channel variables. These corrections are provided from another source, that is, a Kalman filter.

$$\begin{aligned}
 & \left[ \begin{array}{c} -\frac{v_z}{R} \\ \delta\theta_x \\ \delta\theta_y \\ \delta h \\ \delta v_x \\ \delta v_y \\ \frac{d}{dt} \delta v_z \end{array} \right] = \left[ \begin{array}{c} 0 \\ -\frac{v_z}{R^2} \\ -\frac{v_z}{R} \\ 0 \\ -v_y \\ 0 \\ g \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} 0 \\ \frac{1}{R} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} 0 \\ 0 \\ 0 \\ 1 \\ -(\rho + 2\Omega)_x \\ (\rho + 2\Omega)_x \\ -2\Omega_z \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} -\frac{1}{R} \\ 0 \\ 0 \\ 0 \\ -f_z \\ f_z \\ 0 \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ -f_y \\ f_y \\ 0 \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -\Omega_z \\ 0 \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \omega_y \\ -\omega_x \end{array} \right] + \\
 & \quad \left[ \begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \psi_x \\ \psi_y \\ \psi_z \end{array} \right]
 \end{aligned}$$

Fig. 5.3 Navigation system error equations using alternative form for velocity error.



**Fig. 5.4 Representative vertical channel mechanization.**

From this figure, the navigation frame baro-inertial vertical channel dynamic equations are written as

$$\dot{h} = v_z^n - C_1(h - h_B) \quad (5.70)$$

$$\begin{aligned} \dot{v}_z^n &= f_z + g_z + (\rho + 2\Omega)_y v_x^n - (\rho + 2\Omega)_x v_y^n \\ &\quad - C_2(h - h_B) - C_3 \int (h - h_B) \, dt \end{aligned} \quad (5.71)$$

### 5.6.1 Feedback Coefficients

The feedback coefficients,  $C_i$  in Eqs. (5.70) and (5.71), are determined from a specification associated with the altitude dynamics characteristic equation. This equation is a linearized form of these equations and is developed in the following.

In the following, the barometric altitude error  $\delta h_B$  is assumed to be a constant with zero time derivative. A perturbation form of the altitude equation becomes

$$\dot{\delta h} = \delta v_z^n - C_1(\delta h - \delta h_B) \quad (5.72)$$

Differentiating this perturbation form with respect to time yields

$$\ddot{\delta h} = \delta \ddot{v}_z^n - C_1 \dot{\delta h} \quad (5.73)$$

Differentiating this equation again yields

$$\ddot{\delta h} = \ddot{\delta v_z^n} - C_1 \ddot{\delta h}$$

or

$$\ddot{\delta h} + C_1 \ddot{\delta h} = \ddot{\delta v_z^n} \quad (5.74)$$

To complete this equation, an equation for the second derivative of velocity perturbation  $\ddot{\delta v_z^n}$  is required. Neglecting the Earth and transport rate terms in Eq. (5.71), the following linear form of the velocity perturbation is obtained:

$$\ddot{\delta v_z^n} = \ddot{\delta f_z} + \ddot{\delta g_z} - C_2(\delta h - \delta h_B) - C_3 \int (\delta h - \delta h_B) dt \quad (5.75)$$

Gravity is approximated by the following inverse-square expression:

$$g_z = -g_{h=0} \left( \frac{R_{h=0}}{R_{h=0} + h} \right)^2 \quad (5.76)$$

The corresponding gravity perturbation is written as

$$\delta g_z = 2 \left( \frac{g_z}{R_{h=0} + h} \right) \delta h \approx \frac{2g}{R} \delta h \quad (5.77)$$

Gravity in Eq. (5.76) is defined for the navigation frames shown in Figs. 3.2 and 3.3. In Problem 4.7, the direction of gravity is reversed from that defined in this equation—hence the different sign in that exercise.

Differentiating the velocity perturbation equation with respect to time, assuming that the specific force error  $\ddot{\delta f_z}$  is a constant, and substituting the equation for gravity perturbation yield the following for the second derivative of the velocity perturbation:

$$\ddot{\delta v_z^n} \approx \frac{2g}{R} \delta h - C_2 \dot{\delta h} - C_3(\delta h - \delta h_B) \quad (5.78)$$

Substituting this equation into the altitude perturbation differential equation and collecting terms yields the following:

$$\ddot{\delta h} + C_1 \ddot{\delta h} + \left( C_2 - \frac{2g}{R} \right) \dot{\delta h} + C_3 \delta h = C_3 \delta h_B \quad (5.79)$$

This equation describes the altitude error (perturbation) dynamics.

This error dynamic equation can be written as the altitude dynamic characteristic equation, expressed in Laplace transform variable form, as

$$s^3 + C_1 s^2 + \left( C_2 - \frac{2g}{R} \right) s + C_3 = 0 \quad (5.80)$$

The roots of this equation are usually specified to have one real root and a complex pair. The real part of the complex pair has the same magnitude as the pair's imaginary parts, and the single real root also has this value. The complex pair is specified to have a damping ratio of 0.707. That specification is expressed as the following factor of the characteristic equation:

$$(s + \lambda)[s + (\lambda + j\lambda)][s + (\lambda - j\lambda)] = 0 \quad (5.81)$$

and, when multiplied out, results in the following polynomial equation:

$$s^3 + 3\lambda s^2 + 4\lambda^2 s + 2\lambda^3 = 0 \quad (5.82)$$

Equating coefficients of powers of  $s$  in Eqs. (5.80) and (5.82) results in the following expressions for the coefficients,  $C_i$  in the preceding altitude perturbation equation, being obtained:

$$C_1 = 3\lambda, \quad C_2 = 4\lambda^2 + \frac{2g}{R}, \quad \text{and} \quad C_3 = 2\lambda^3 \quad (5.83)$$

The value of  $\lambda$  is usually specified as 0.01.

### 5.6.2 Instability

The instability in the vertical channel can be seen from the altitude perturbation equation (5.79) with the  $C_i$  coefficients set to zero. In this case, the roots of this equation are zero and a real pair with roots  $\pm\sqrt{2g/R}$ . The positive root is a source of instability for the vertical channel.

## 5.7 Chapter Summary

The development of equations for a strap-down navigation system, describing the evolution of the navigation states of position, velocity, and attitude, were presented. In developing these equations, the objective was to establish these equations in terms of sensed accelerations and rates, as provided by corresponding sets of next three orthogonal sensors. Examples of inertial rate sensors are described next in Chapter 6.

Different forms of linearized error equations were developed as design alternatives, each with their advantages and disadvantages. These illustrations of different forms to describe navigation system errors should reinforce the notion that an assumed error model needs to be verified prior to attempting to integrate other elements within an existing navigation system. All of the error models presented in this chapter can be found in the literature. Examples of these different linearized forms are used in several integrated navigation system applications to be presented later in Part 2.

---

## Problems

- 5.1** This exercise concerns inertial navigation frame mechanization equations. Paralleling the developments in this chapter, obtain the following equations describing position and velocity in an inertial frame:

Position:

$$\vec{r}^i = C_b^i f^b + \mathbf{G}^i$$

Velocity:

$$\mathbf{v}^i = \dot{\mathbf{r}}^i$$

and find that the attitude equations simplify to the following for a strap-down inertial sensor configuration:

Attitude:

$$\dot{C}_b^i = C_b^i \Omega_{i/b}^b$$

where the skew-symmetric matrix  $\Omega_{i/b}^b$  is made up of the gyro-sensed angular rotation vector in the body frame.

- 5.2** This exercise concerns inertial navigation frame error equations. Assume that equations in the preceding problem hold for both true and computed position, velocity, and attitude. Use the error representations

$$\bar{\mathbf{r}}^i = \mathbf{r}^i + \delta \mathbf{r}^i$$

$$\bar{\mathbf{f}}^i = \bar{C}_b^i \bar{\mathbf{f}}^b = [\mathbf{I} - (\boldsymbol{\phi} \times)] C_b^i (f^b + \delta f^b) \approx [\mathbf{I} - (\boldsymbol{\phi} \times)] f^i + \delta \mathbf{f}^i$$

$$\bar{\mathbf{G}}^i = \mathbf{G}^i + \delta \mathbf{G}^i$$

to obtain the following error equation for position/velocity:

$$\delta \ddot{\mathbf{r}}^i = \mathbf{f}^i \times \boldsymbol{\phi} + \delta \mathbf{f}^i + \delta \mathbf{G}^i$$

where the gravitation error is a function of position error and was shown earlier in Problem 4.6 to be

$$\delta \mathbf{G}^i = \omega^2 [3\mathbf{e} \mathbf{e}^T - \mathbf{I}] \delta \mathbf{r}^i$$

to yield the following for the position/velocity error equation:

$$\delta \ddot{\mathbf{r}}^i = \omega^2 [3\mathbf{e} \mathbf{e}^T - \mathbf{I}] \delta \mathbf{r}^i + \mathbf{f}^i \times \boldsymbol{\phi} + \delta \mathbf{f}^i$$

Show that tilt error dynamics are given as

$$\dot{\phi} = \epsilon^i$$

and that position/velocity and attitude error equations can be expressed in the following summary form:

$$\frac{d}{dt} \begin{bmatrix} \delta r^i \\ \delta v^i \\ \boldsymbol{\phi} \end{bmatrix} = \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \\ \omega^2 [3\mathbf{e}\mathbf{e}^T - I] & 0_{3 \times 3} & (\mathbf{f}^i \times) \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} \begin{bmatrix} \delta r^i \\ \delta v^i \\ \boldsymbol{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ \delta f^i \\ \boldsymbol{\epsilon}^i \end{bmatrix}$$

- 5.3** Use alternate velocity error representation and linear position errors to obtain navigation system error equations, summarized next, for a wander-azimuth terrestrial navigation system (assume  $\delta\theta_z \equiv 0$ ). The linear position error is related to the angular position error by the following:

$$\delta\theta \equiv \begin{bmatrix} -\frac{\delta r_y}{R} \\ \frac{\delta r_x}{R} \\ \delta\theta_z \end{bmatrix}$$

*Hint:* From Eq. (5.66),

$$\delta\dot{\theta}_x = -\frac{v_z}{R} \delta\theta_x + \frac{v_y}{R^2} \delta h - \frac{l}{R} \delta v_y^l$$

With the  $x$  component of angular position error defined as just shown and taking its derivative, the following also holds:

$$\delta\dot{\theta}_x = -\frac{l}{R} \delta\dot{r}_y - \frac{v_z}{R} \delta\theta_x$$

Equating these two equations for  $\delta\dot{\theta}_x$  yields the following equation for the  $y$  component of linear position error:

$$\delta\dot{r}_y = -\frac{v_y}{R} \delta h + \delta v_y^l$$

Or, using the transport rate vector as defined in Eq. (5.25),

$$\delta\dot{r}_y = \rho_x \delta h + \delta v_y^l$$

The remaining terms can be obtained similarly and are summarized as follows:

$$\begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta h \\ \delta v_x^n \\ \delta v_y^n \\ \phi_x \\ \phi_y \\ \phi_z \end{bmatrix} = \frac{d}{dt} \begin{bmatrix} 0 & 0 & -\rho_y & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \rho_x & 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{\rho_y}{R} & -\rho_x & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\Omega_z & -(\rho + 2\Omega)_y & 0 & -f_z & f_y \\ \frac{(f_z - g)}{R} & 0 & 0 & -2\Omega_z & -\frac{v_z}{R} & (\rho + 2\Omega)_x & f_z & 0 & -f_x \\ -\frac{f_x}{R} & \frac{(f_z - g)}{R} & 0 & -2\Omega_z & -\frac{2g}{R} & (\omega + \Omega)_y & 0 & -f_y & f_x \\ \frac{\Omega_z}{R} & -\frac{f_y}{R} & -\frac{\rho_x}{R} & 0 & -\frac{1}{R} & 0 & 0 & 0 & -\omega_y \\ -\frac{v_z}{R^2} & \frac{\Omega_z}{R^2} & -\frac{\rho_y}{R} & 0 & -\frac{1}{R} & 0 & 0 & \Omega_z & -\omega_y \\ -\frac{v_z}{R^2} & -\frac{\Omega_z}{R} & -\frac{\rho_y}{R} & 0 & \frac{1}{R} & 0 & 0 & -\Omega_z & 0 \\ \frac{\omega_x}{R} & 0 & 0 & 0 & 0 & 0 & \omega_y & -\omega_x & 0 \end{bmatrix} \times \begin{bmatrix} \frac{\partial r_x}{\partial r_y} \\ \frac{\partial r_y}{\partial h} \\ \frac{\partial v_x^n}{\partial r_x} \\ \frac{\partial v_y^n}{\partial r_y} \\ \phi_x \\ \phi_y \\ \phi_z \end{bmatrix}$$

- 5.4** This exercise concerns attitude error equations for ECEF frame. Parallel the developments in this section to obtain the following equations describing attitude error dynamics for the ECEF frame. Assume the computed body to ECEF frame transformation matrix is represented as

$$\bar{C}_b^e \equiv [I - (\boldsymbol{\Psi} \times)] C_b^e$$

Then the error in this matrix is

$$\bar{C}_b^e - C_b^e = \delta C_b^e = -(\boldsymbol{\Psi} \times) C_b^e$$

Using the following expression for the dynamics of this matrix:

$$\dot{C}_b^e = -\Omega_{b/e}^e C_b^e$$

obtain the following differential equation for  $\boldsymbol{\Psi}$  attitude error dynamics:

$$\dot{\boldsymbol{\Psi}} = \boldsymbol{\Psi} \times \boldsymbol{\omega}_{b/e}^e + (\bar{\boldsymbol{\omega}}_{b/e}^e - \boldsymbol{\omega}_{b/e}^e)$$

Then, using the following expressions for the rotation vector  $\boldsymbol{\omega}_{b/e}^e$ :

$$\bar{\boldsymbol{\omega}}_{b/e}^e = \bar{\boldsymbol{\omega}}_{i/e}^e - \bar{C}_b^e \boldsymbol{\omega}_{i/b}^b + \boldsymbol{\varepsilon}^e$$

$$\boldsymbol{\omega}_{b/e}^e = \boldsymbol{\omega}_{i/e}^e - C_b^e \boldsymbol{\omega}_{i/b}^b$$

obtain the following final form:

$$\dot{\boldsymbol{\Psi}} = \boldsymbol{\Psi} \times \boldsymbol{\omega}_{i/e}^e + \boldsymbol{\varepsilon}^e$$

where Earth rate error is assumed zero

$$\delta \boldsymbol{\omega}_{i/e}^e = 0$$

- 5.5** Obtain Eq. (5.1) by considering the following. First, transform the position vector from an inertial frame to an ECEF frame as

$$\mathbf{r}^e = C_i^e \mathbf{r}^i$$

Take the time derivative to obtain

$$\dot{\mathbf{r}}^e = -(\boldsymbol{\omega}_{i/e}^e \times) \mathbf{r}^e + C_i^e \dot{\mathbf{r}}^i$$

or because

$$\begin{aligned} \dot{\mathbf{r}}^i &\equiv \mathbf{V}^i \quad \text{and} \quad \mathbf{V}^e = C_i^e \mathbf{V}^i, \\ \dot{\mathbf{r}}^e &= -(\boldsymbol{\omega}_{i/e}^e \times) \mathbf{r}^e + \mathbf{V}^e \end{aligned}$$

Define a velocity of the local position as a result of the Earth's rotation

$$\mathbf{V}_p^e = \boldsymbol{\omega}_{i/e}^e \times \mathbf{r}^e$$

then the velocity relative to this position

$$\begin{aligned}\mathbf{v}^e &= \mathbf{V}^e - \mathbf{V}_p^e \\ &= \mathbf{V}^e - (\boldsymbol{\omega}_{i/e}^e \times \mathbf{r}^e)\end{aligned}$$

The relative velocity is identical to the right-hand side of the preceding position derivative equation:

$$\begin{aligned}\dot{\mathbf{r}}^e &= \mathbf{v}^e \\ &= C_n^e \mathbf{v}^n\end{aligned}$$

or

$$\mathbf{v}^n = C_e^n \dot{\mathbf{r}}^e$$

- 5.6** Use the angular velocity addition theorem from Sec. 2.5 to obtain an expression for the body-referenced body-to-geographic frame rotation rate vector  $\boldsymbol{\omega}_{g/b}^b$  in terms of Euler angle rates  $\dot{\phi}$ ,  $\dot{\theta}$ , and  $\dot{\psi}$  for a geographic north–east–down (*N*–*E*–*D*) navigation frame and a body frame whose *x*–*y*–*z* axes are parallel to the *N*–*E*–*D* directions when Euler angles are all zero.

*Hint:* The following transformation matrix transforms geographic frame vector components to the body frame following a yaw–pitch–roll sequence of rotations:

$$\begin{aligned}C_g^b &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\varphi & s\varphi \\ 0 & -s\varphi & c\varphi \end{bmatrix} \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix} \begin{bmatrix} c\psi & s\psi & 0 \\ -s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\varphi s\theta c\psi - c\varphi s\psi & s\varphi s\theta s\psi + c\varphi c\psi & s\varphi c\theta \\ c\varphi s\theta c\psi + s\varphi s\psi & c\varphi s\theta s\psi - s\varphi c\psi & c\varphi c\theta \end{bmatrix}\end{aligned}$$

Show the following:

$$\begin{aligned}\boldsymbol{\omega}_{g/b}^b &= \begin{bmatrix} P \\ Q \\ R \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\varphi & s\varphi \\ 0 & -s\varphi & c\varphi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} c\theta & 0 & -s\theta \\ s\varphi s\theta & c\varphi & s\varphi c\theta \\ c\varphi s\theta & -s\varphi & c\varphi c\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\varphi & s\varphi c\theta \\ 0 & -s\varphi & c\varphi c\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}\end{aligned}$$

Then, the equation for Euler angle rates is obtained as

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = R^{-1} \boldsymbol{\omega}_{g/b}^b = \begin{bmatrix} 1 & s\varphi \frac{s\theta}{c\theta} & c\varphi \frac{s\theta}{c\theta} \\ 0 & c\varphi & -s\varphi \\ 0 & s\varphi \frac{1}{c\theta} & c\varphi \frac{1}{c\theta} \end{bmatrix} \boldsymbol{\omega}_{g/b}^b$$

- 5.7** Use the result in Problem 5.6 to obtain equations for Euler-angle rates in terms of navigation frame and gyro rates.

*Hint:* The body-frame-referenced body-to-local-geographic frame rotation rate vector  $\boldsymbol{\omega}_{g/b}^b$  can be expressed, using the angular velocity addition theorem, as

$$\begin{aligned} \boldsymbol{\omega}_{g/b}^b &= \boldsymbol{\omega}_{g/i}^b + \boldsymbol{\omega}_{i/b}^b \\ &= \boldsymbol{\omega}_{i/b}^b - \boldsymbol{\omega}_{i/g}^b \\ &= \boldsymbol{\omega}_{i/b}^b - C_g^b \boldsymbol{\omega}_{i/g}^g \end{aligned}$$

In this equation, the preceding body-frame-referenced body-to-inertial frame rotation rate vector  $\boldsymbol{\omega}_{i/b}^b$  (gyro output rates) is denoted as

$$\boldsymbol{\omega}_{i/b}^b \equiv \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

and the local geographic-frame-referenced geographic-to-inertial frame rotation rate vector is composed of “Earth” and “transport” rates and rewritten in the following notation:

$$\begin{aligned} \boldsymbol{\omega}_{i/g}^g &= \boldsymbol{\omega}_{i/e}^g + \boldsymbol{\omega}_{e/g}^g \\ &\equiv \boldsymbol{\Omega} + \boldsymbol{\rho} \end{aligned}$$

Euler-angle rates can be expressed as

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = R^{-1} \begin{bmatrix} p \\ q \\ r \end{bmatrix} - R^{-1} C_g^b (\boldsymbol{\Omega} + \boldsymbol{\rho})$$

Express the matrix  $R^{-1}$  as the following product:

$$R^{-1} = \begin{bmatrix} 1 & 0 & \frac{s\theta}{c\theta} \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{c\theta} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\varphi & -s\varphi \\ 0 & s\varphi & c\varphi \end{bmatrix}$$

demonstrate that the product  $R^{-1}C_g^b$  becomes

$$R^{-1}C_g^b = \begin{bmatrix} \frac{c\psi}{c\theta} & \frac{s\psi}{c\theta} & 0 \\ -s\psi & c\psi & 0 \\ \frac{s\theta}{c\theta}c\psi & \frac{s\theta}{c\theta}s\psi & 1 \end{bmatrix}$$

Returning to the last term in the preceding Euler-angle rates equation and using the result just obtained, obtain the following:

$$\begin{bmatrix} \omega_\varphi \\ \omega_\theta \\ \omega_\psi \end{bmatrix} \equiv R^{-1}C_g^b(\boldsymbol{\Omega} + \boldsymbol{\rho})$$

$$= \begin{bmatrix} \frac{c\psi}{c\theta} & \frac{s\psi}{c\theta} & 0 \\ -s\psi & c\psi & 0 \\ \frac{s\theta}{c\theta}c\psi & \frac{s\theta}{c\theta}s\psi & 1 \end{bmatrix} \begin{bmatrix} \Omega_n + \rho_n \\ \rho_e \\ \Omega_d + \rho_d \end{bmatrix}$$

And, for each component,

$$\begin{aligned} \omega_\varphi &= \frac{c\psi}{c\theta}(\Omega_n + \rho_n) + \frac{s\psi}{c\theta}\rho_e \\ \omega_\theta &= -s\psi(\Omega_n + \rho_n) + c\psi\rho_e \\ \omega_\varphi &= \frac{s\theta}{c\theta}c\psi(\Omega_n + \rho_n) + \frac{s\theta}{c\theta}s\psi\rho_e + (\Omega_d + \rho_d) \end{aligned}$$

- 5.8** This exercise concerns position difference observation and linearization. In the following, an observation is formed from vector dot products of columns of the wander-azimuth frame direction cosine matrix  $C_n^e(x)$ , containing error, and another column vector formed from the geodetic referenced position, that is, latitude and longitude, without error. The results of the vector dot products are the wander-azimuth angular position difference that can be used as position observations in a navigation Kalman filter.

Consider the following form for the computed  $C_n^e(x)$  direction cosine matrix:

$$\tilde{C}_n^e(x) = \begin{bmatrix} c\bar{\alpha}c\bar{\phi} & -s\bar{\alpha}c\bar{\phi} & s\bar{\phi} \\ c\bar{\alpha}s\bar{\phi}s\bar{\lambda} + s\bar{\alpha}c\bar{\lambda} & -s\bar{\alpha}s\bar{\phi}s\bar{\lambda} + c\bar{\alpha}c\bar{\lambda} & -c\bar{\phi}s\bar{\lambda} \\ -c\bar{\alpha}s\bar{\phi}c\bar{\lambda} + s\bar{\alpha}s\bar{\lambda} & s\bar{\alpha}s\bar{\phi}c\bar{\lambda} + c\bar{\alpha}s\bar{\lambda} & c\bar{\phi}c\bar{\lambda} \end{bmatrix}$$

where each of the elements are assumed to be composed of the true value plus an error as

$$\begin{aligned}\bar{\phi} &= \phi + \delta\phi \\ \bar{\lambda} &= \lambda + \delta\lambda \\ \bar{\alpha} &= \alpha + \delta\alpha\end{aligned}$$

This matrix can be rewritten as

$$\bar{C}_n^e \equiv [\bar{s}_1 | \bar{s}_2 | \bar{s}_3]$$

where the unit vectors  $s_i$  represent the columns indicated. Because the direction cosine matrix is orthonormal, these unit vectors are normal to each other. That is, the vector dot products formed from these column vectors are zero.

The last column of  $C_n^e(x)$ , expressed in terms of the true values (without error), is

$$t_3 = \begin{bmatrix} s\phi \\ -c\phi s\lambda \\ c\phi c\lambda \end{bmatrix}$$

Taking the dot product of this vector with the first two column vectors of  $\bar{C}_n^e$  yields the following results:

$$\begin{aligned}\bar{s}_1 \cdot t_3 &= -c\alpha\delta\phi + s\alpha c\phi\delta\lambda \\ \bar{s}_2 \cdot t_3 &= s\alpha\delta\phi + c\alpha c\phi\delta\lambda\end{aligned}$$

It was shown in Example 3.3 that the right-hand sides of these equations are equivalent to the  $y$  and  $x$  angular position errors, respectively. Therefore,

$$\begin{aligned}\delta\theta_x &= \bar{s}_2 \cdot t_3 \\ \delta\theta_y &= -\bar{s}_1 \cdot t_3\end{aligned}$$

The first few elements corresponding to the system errors of the linearized measurement matrix become

$$H = \frac{\partial \mathbf{h}(x, t)}{\partial \delta \boldsymbol{\theta}} = \begin{bmatrix} 1 & 0 & \dots \\ 0 & 1 & \dots \end{bmatrix}$$

- 5.9** In the following, express the difference between the navigation solution's velocity (in the wander-azimuth frame), as provided by an inertial navigation unit (INU), and the geodetic-frame-referenced GPS velocity, and show that this difference depends on position errors.

Consider the difference

$$\begin{aligned}\Delta \mathbf{v}^n &= \bar{\mathbf{v}}_{\text{INU}}^n - \bar{C}_g^n \mathbf{v}_{\text{GPS}}^g \\ &= \mathbf{v}^n + \delta \mathbf{v}^n - \begin{bmatrix} c\bar{\alpha} & s\bar{\alpha} & 0 \\ -s\bar{\alpha} & c\bar{\alpha} & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{v}^g \\ &= \delta \mathbf{v}^n + \begin{bmatrix} -v_y^n \delta \alpha \\ v_x^n \delta \alpha \\ 0 \end{bmatrix}\end{aligned}$$

The wander-azimuth error  $\delta \alpha$  can be related to the angular position error vector components as

$$\begin{aligned}\delta \alpha &= \delta \theta_z - \Delta \lambda s \phi \\ &= \delta \theta_z - t \phi (c \alpha \Delta \theta_x - s \alpha \Delta \theta_y)\end{aligned}$$

Substituting this expression into the preceding equation yields the following:

$$\begin{aligned}\Delta \mathbf{v}^n &= \delta \mathbf{v}^n + \begin{bmatrix} -[\delta \theta_z - t \phi (c \alpha \delta \theta_x - s \alpha \delta \theta_y)] v_y^n \\ [\delta \theta_z - t \phi (c \alpha \delta \theta_x - s \alpha \delta \theta_y)] v_x^n \\ 0 \end{bmatrix} \\ &= \delta \mathbf{v}^n + \begin{bmatrix} t \phi c \alpha v_y^n & -t \phi s \alpha v_y^n & -v_y^n \\ -t \phi c \alpha v_x^n & t \phi s \alpha v_x^n & v_x^n \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \theta_x \\ \delta \theta_y \\ \delta \theta_z \end{bmatrix}\end{aligned}$$

- 5.10** A comparison of velocities from two noncollocated sources fixed to a rigid body must consider their relative displacements and the body angular motion that induces additional velocity. In this exercise, GPS velocities referenced at the antenna and INU velocities will be used to establish their navigation-frame velocity difference.

The INU's position vector  $\mathbf{r}_{\text{INU}}$  is related to the GPS antenna position vector  $\mathbf{r}_{\text{ant}}$ , and lever arm  $\mathbf{d}$  position vector by

$$\mathbf{r}_{\text{INU}} = \mathbf{r}_{\text{ant}} + \mathbf{d}$$

Representing this vector in an Earth-fixed reference frame,

$$\begin{aligned}\mathbf{r}_{\text{INU}}^e &= \mathbf{r}_{\text{ant}}^e + \mathbf{d}^e \\ &= \mathbf{r}_{\text{ant}}^e + C_n^e C_b^n \mathbf{d}^b\end{aligned}$$

Use the prior relationships between the time derivative of these ECEF position vectors and velocity to form navigation-frame-referenced velocity differences. Take the time derivative of this equation,

recognizing that the lever arm is fixed (a constant) in the body frame, to show

$$\begin{aligned}\dot{\mathbf{r}}_{\text{INU}}^e &= \dot{\mathbf{r}}_{\text{ant}}^e + \dot{C}_n^n C_b^n \mathbf{d}^b + C_n^n \dot{C}_b^n \mathbf{d}^b \\ &= \dot{\mathbf{r}}_{\text{ant}}^e + C_n^n \Omega_{e/n}^n C_b^n \mathbf{d}^b + C_n^n \Omega_{n/b}^n C_b^n \mathbf{d}^b\end{aligned}$$

Recall that the velocity in the local-level navigation frame is defined as

$$\mathbf{v}^n \equiv C_e^n \dot{\mathbf{r}}^e$$

and, after premultiplying the  $\dot{\mathbf{r}}^e$ -equation by  $C_n^n$ , show that

$$\begin{aligned}\mathbf{v}_{\text{INU}}^n &= \mathbf{v}_{\text{ant}}^n + (\Omega_{e/n}^n + \Omega_{n/b}^b) C_b^n \mathbf{d}^b \\ &= \mathbf{v}_{\text{ant}}^n + [(\boldsymbol{\omega}_{e/n}^n + \boldsymbol{\omega}_{n/b}^b) \times] C_b^n \mathbf{d}^b\end{aligned}$$

This equation shows the additional velocity at the antenna location.

- 5.11** This exercise establishes the state vector form of the vertical channel error dynamics suitable for a Kalman filter implementation. Rewrite Eq. (5.75) as

$$\delta \dot{v}_z^n \approx \delta f_z + \delta g_z - C_2(\delta h - \delta h_B) + \delta a$$

where the equation for  $\delta a$  is given by

$$\delta \dot{a} = -C_3(\delta h - \delta h_B)$$

Express these error equations in the following state vector form:

$$\frac{d}{dt} \begin{bmatrix} \delta h \\ \delta v_z^n \\ \delta a \\ \delta h_B \\ \delta f_z \end{bmatrix} = \begin{bmatrix} -C_1 & 1 & 0 & C_1 & 0 \\ -\left(C_2 - \frac{2g}{R}\right) & 0 & 1 & C_2 & 1 \\ -C_3 & 0 & 0 & C_3 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta h \\ \delta v_z^n \\ \delta a \\ \delta h_B \\ \delta f_z \end{bmatrix} + \begin{bmatrix} 0 \\ w_{v_z} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

In Sec. 5.6, there are modeling idealizations in the vertical velocity error dynamics, for example, transport and Earth rate terms and the attitude error terms are neglected. To recognize that these contributors are not included in the system error dynamics, process noise  $w_{v_z}$  is included in the velocity error state to account for these missing errors.

- 5.12** Obtaining the vector form in Eq. (5.34) from Eq. (5.33) [and Eq. (5.51) from Eq. (5.50)] is the subject of this exercise. Consider the expanded

skew-symmetric differential equation from Eq. (5.33):

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix} &= - \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix} \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & -\delta\omega_z & \delta\omega_y \\ \delta\omega_z & 0 & -\delta\omega_x \\ -\delta\omega_y & \delta\omega_x & 0 \end{bmatrix} \\ &= - \begin{bmatrix} \dots & \omega_y\phi_x & \omega_z\phi_x \\ \dots & \dots & \omega_z\phi_y \\ \dots & \dots & \dots \end{bmatrix} + \begin{bmatrix} \dots & \phi_y\omega_x & \phi_z\omega_x \\ \dots & \dots & \phi_z\omega_y \\ \dots & \dots & \dots \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & -\delta\omega_z & \delta\omega_y \\ \delta\omega_z & 0 & -\delta\omega_x \\ -\delta\omega_y & \delta\omega_x & 0 \end{bmatrix} \end{aligned}$$

Collect terms by matrix element:

$$\begin{aligned} 1:2 \quad -\dot{\phi}_z &= -\omega_y\phi_x + \omega_x\phi_y - \delta\omega_z \\ 1:3 \quad \dot{\phi}_y &= -\omega_z\phi_x + \omega_x\phi_z + \delta\omega_y \\ 2:3 \quad -\dot{\phi}_x &= -\omega_z\phi_y + \omega_y\phi_z - \delta\omega_x \end{aligned}$$

And rearrange to form the vector equation

$$\frac{d}{dt} \begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix} = - \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix} + \begin{bmatrix} \delta\omega_x \\ \delta\omega_y \\ \delta\omega_z \end{bmatrix}$$

Rewrite this as a vector differential equation

$$\begin{aligned} \dot{\phi} &= -\omega\phi + \delta\omega \\ &= \phi\omega + \delta\omega \end{aligned}$$

This is Eq. (5.34).

- 5.13** Next is an exception to the statement regarding the general application of the position error dynamic equation. The *N-E-D* navigation (geographic) frame requires only two error angles to express error in the

DCM as shown next. The DCM from Chapter 3 is

$$C_e^g(z) = \begin{bmatrix} -s\varphi c\lambda & -s\varphi s\lambda & c\varphi \\ -s\lambda & c\lambda & 0 \\ -c\varphi c\lambda & -c\varphi s\lambda & -s\varphi \end{bmatrix}$$

Defining errors in latitude and longitude by

$$\bar{\varphi} = \varphi + \delta\varphi \quad \text{and} \quad \bar{\lambda} = \lambda + \delta\lambda$$

and using small-angle approximations

$$s\bar{\varphi} \approx s\varphi + c\varphi\delta\varphi, \quad c\bar{\varphi} \approx c\varphi - s\varphi\delta\varphi \quad \text{etc.}$$

then, the corrupted form of the DCM can be related to the uncorrupted form as follows:

$$\begin{aligned} \bar{C}_e^g &= \begin{bmatrix} -s\bar{\varphi}c\bar{\lambda} & -s\bar{\varphi}s\bar{\lambda} & c\bar{\varphi} \\ -s\bar{\lambda} & c\bar{\lambda} & 0 \\ -c\bar{\varphi}c\bar{\lambda} & -c\bar{\varphi}s\bar{\lambda} & -s\bar{\varphi} \end{bmatrix} \\ &= \begin{bmatrix} -(s\varphi + c\varphi\delta\varphi)(c\lambda - s\lambda\delta\lambda) & \dots & (c\varphi - s\varphi\delta\varphi) \\ -(s\lambda + c\lambda\delta\lambda) & (c\lambda - s\lambda\delta\lambda) & 0 \\ -(c\varphi - s\varphi\delta\varphi)(c\lambda - s\lambda\delta\lambda) & \dots & -(s\varphi + c\varphi\delta\varphi) \end{bmatrix} \\ &\approx \begin{bmatrix} -s\varphi c\lambda - c\varphi c\lambda\delta\varphi + s\varphi s\lambda\delta\lambda & \dots & c\varphi - s\varphi\delta\varphi \\ -s\lambda - c\lambda\delta\lambda & c\lambda - s\lambda\delta\lambda & 0 \\ -c\varphi c\lambda + s\varphi c\lambda\delta\varphi + c\varphi s\lambda\delta\lambda & \dots & -s\varphi - c\varphi\delta\varphi \end{bmatrix} \\ &= \begin{bmatrix} -s\varphi c\lambda & -s\varphi s\lambda & c\varphi \\ -s\lambda & c\lambda & 0 \\ -c\varphi c\lambda & -c\varphi s\lambda & -s\varphi \end{bmatrix} \\ &+ \begin{bmatrix} 0 & -s\varphi\delta\lambda & \delta\varphi \\ s\varphi\delta\lambda & 0 & c\varphi\delta\lambda \\ -\delta\varphi & -c\varphi\delta\lambda & 0 \end{bmatrix} \begin{bmatrix} -s\varphi c\lambda & -s\varphi s\lambda & c\varphi \\ -s\lambda & c\lambda & 0 \\ -c\varphi c\lambda & -c\varphi s\lambda & -s\varphi \end{bmatrix} \\ &\equiv C_e^g - (\delta\theta^g x) C_e^g \\ &= [I - (\delta\theta^g x)] C_e^g \end{aligned}$$

where the angular position error vector from this approach would be defined as

$$\delta\theta^g = \begin{bmatrix} c\varphi\delta\lambda \\ -\delta\varphi \\ -s\varphi\delta\lambda \end{bmatrix}$$

Demonstrate that the problem in using this form in the position dynamics equations developed in the text arises from the fact that the elements in

$$\left[ \begin{array}{c} 0 & 0 & -\frac{v_n}{R^2} & \frac{1}{R} & 0 & 0 & 0 & 0 \\ \frac{v_e t \varphi}{R c \varphi} & 0 & -\frac{v_e}{R^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{R^2 c \varphi}{R^2 c \varphi} & 0 & \frac{1}{R c \varphi} & 0 & 0 & 0 \\ -\frac{v_e^2}{R c^2 \varphi} - 2 \omega_{j/e} c \varphi v_e & 0 & -\frac{v_n v_d}{R^2} + \frac{v_e^2 t \varphi}{R^2} & \frac{v_d}{R} & 0 & -1 & 0 & 0 \\ \frac{v_n v_e}{R c^2 \varphi} + 2 \omega_{j/e} (c \varphi v_n - s \varphi v_e) & 0 & -\frac{v_e v_d}{R^2} - \frac{v_n v_e t \varphi}{R^2} & -(\rho + 2\omega)_d & \frac{2(\rho + \omega)_d}{R} & -\rho_e & 0 & 0 \\ \frac{\mathrm{d}}{\mathrm{d}t} & \frac{\delta \varphi}{\delta \lambda} & 0 & -\frac{(v_n^2 + v_e^2)}{R^2} & -2(\rho + \omega)_n & (\rho + 2\omega)_n & f_e & -f_n \\ \phi_n & -\omega_{j/e} s \varphi & 0 & \frac{2\rho_e}{R^2} & -\frac{1}{R} & 0 & -f_e & f_n \\ \phi_e & 0 & 0 & -\frac{v_e}{R^2} & 0 & 0 & (\rho + \omega)_d & -\rho_e \\ \phi_d & 0 & 0 & \frac{v_n}{R^2} & -\frac{1}{R} & 0 & -(\rho + \omega)_d & (\rho + \omega)_n \\ \phi_u & -\omega_{j/e} c \varphi - \frac{v_e}{R c^2 \varphi} & 0 & \frac{v_e t \varphi}{R^2} & 0 & 0 & -(\rho + \omega)_n & 0 \end{array} \right] + \left[ \begin{array}{c} \delta \varphi \\ \delta \lambda \\ \delta h \\ \delta v_n \\ \delta v_d \\ \delta \omega_e \\ \delta \omega_d \\ \delta \phi_u \end{array} \right]$$

this vector are linearly related:

$$s\varphi\delta\theta_n^g + c\varphi\delta\theta_d^g = 0$$

Therefore, only two error angles are required to describe error in the DCM above which are the two original angles for latitude and longitude error.

- 5.14.** Establish the following “ $\phi$ ” form error dynamic model based on latitude and longitude position errors starting with the position equations in Example 4.1 and using Eqs. (5.24) and (5.43).

## Navigation Sensor Models

In integrated navigation systems analysis, inertial sensors are described by their performance characteristics and error models. In most integrated navigation system implementations the gyro's characteristics are of primary importance. This presentation of navigation sensor models includes the following: gyro performance characterizations and sensor error models.

Inertial sensors commonly used in navigation units are rate/rate integrating gyros, dynamically tuned rotor gyros, ring laser gyros (RLG), fiber-optic gyros (FOG), and various types of accelerometers. The most recent developments are in RLG and FOG gyro technologies.

Both RLG and FOG sensors employ the Sagnac effect. This effect is described at a simplified level in this chapter. These sensors are presented to illustrate how the gyro's design, that is, RLG path length, influences its performance.

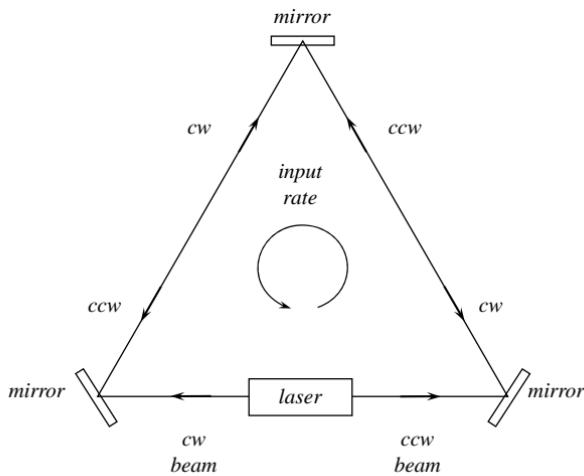
A vital part of the design and evaluation of integrated navigation systems is the ability to model and simulate errors associated with gyros and accelerometers. Models allow computer-based evaluations prior to fabricating hardware, thus saving time and money. General mathematical models representing gyro and accelerometer errors are presented. Terms included within these representations are long-term stable errors, which can be described as random constants, and short-term random errors, represented as time-correlated and random walk errors. Dynamic models for time-correlated and random walk errors are developed, with examples from simulations illustrating the model's time characteristics.

Problems are included that expand upon the material presented.

### 6.1 Gyro Performance Characterizations

#### 6.1.1 Ring Laser Gyro

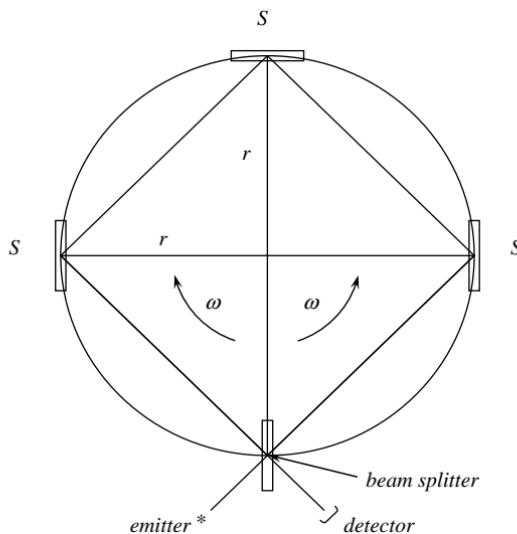
The RLG has recently seen increased usage in strap-down navigation system mechanizations. Most current RLG sensors are single degree-of-freedom sensors requiring three of these for an inertial navigation system implementation, as described in Chapter 5. A single degree-of-freedom RLG is shown schematically in Fig. 6.1. This figure illustrates a triangular version on the RLG. Other versions exist with four sides. The gyro includes a laser, a closed-path cavity, mirrors at each intermediate corner in the path, and an interferometer/photodetector. The operation of the gyro is based on optical and electronic phenomena, rather



**Fig. 6.1** Ring laser gyro.

than the mechanical phenomena. As a result, these gyros exhibit little of the error associated with acceleration, as with mechanical gyro.

The optical phenomenon employed by this gyro is the Sagnac effect. Figure 6.2 is an illustration of Sagnac's experiment [8]. Two laser beams, from the same source, propagate around the closed path in opposite directions: clockwise *cw* and counterclockwise *ccw*. Three mirrors, indicated with an *S*, with a beam splitter form a square enclosed in a circle of radius *r*.



**Fig. 6.2** Sagnac experiment.

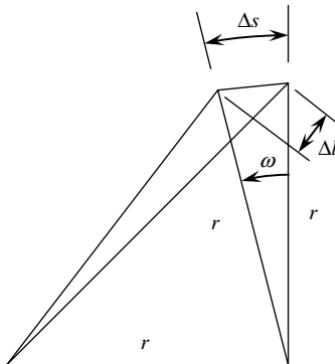


Fig. 6.3 Sagnac experiment: one leg.

This effect can be described based on geometrical optics. Shown in Fig. 6.3 is one leg of the experimental arrangement. As a result of the rotation  $\omega$ , the effective path length traveled in the direction shown is shortened. Assuming small angles, the effective path of the leg shown is shortened by

$$\begin{aligned}\Delta l &= \frac{\sqrt{2}}{2} \Delta s \\ &= \frac{\sqrt{2}}{2} r \omega \tau\end{aligned}\quad (6.1)$$

for a rotation duration  $\tau$ . The time to propagate around the closed path in one direction is

$$\tau_+ = \frac{(l + \Delta l)}{c} \quad (6.2)$$

and, in the opposite direction, it is

$$\tau_- = \frac{(l - \Delta l)}{c} \quad (6.3)$$

where  $c$  is the light propagation speed. The difference in time to propagate in the two directions is

$$\begin{aligned}\Delta \tau &= \tau_+ - \tau_- \\ &= 2 \frac{\Delta l}{c} \\ &= \sqrt{2} \frac{r \omega \tau}{c}\end{aligned}\quad (6.4)$$

Define the preceding rotation period as the time to transient around the closed path without rotation:

$$\tau = 4\sqrt{2} \frac{r}{c} \quad (6.5)$$

Then, the time difference between the two propagation directions becomes

$$\Delta\tau = 8 \frac{r^2 \omega}{c^2} \quad (6.6)$$

The time difference can be related to a change in wavelength  $\lambda$  by

$$\frac{\Delta\lambda}{\lambda} = \frac{\Delta\tau}{\tau} \quad (6.7)$$

and the corresponding wavelength difference becomes

$$\Delta\lambda = \sqrt{2} \frac{r\omega\lambda}{c} \quad (6.8)$$

For the geometry illustrated, the enclosed area  $A$  formed by the path and its perimeter  $P$  are

$$A = 2r^2$$

$$P = 2\sqrt{2}r$$

such that the change in wavelength can be expressed as

$$\Delta\lambda = \sqrt{2} \frac{r\omega A}{Pc/\lambda} \quad (6.9)$$

The frequency is related to the wavelength by

$$\nu = \frac{c}{\lambda} \quad (6.10)$$

or

$$\nu\lambda = c \quad (6.11)$$

Because  $c$  is a constant, the variation of Eq. (6.11) yields

$$\begin{aligned} \Delta\nu &= -\frac{\Delta\lambda}{\lambda}\nu \\ &= -\frac{4\omega A}{Pc}\nu \end{aligned} \quad (6.12)$$

or, from Eq. (6.10),

$$|\Delta\nu| = \frac{4\omega A}{P\lambda} \quad (6.13)$$

This result shows that the frequency change as a result of the gyro's rotation is more sensitive with the higher ratio of the area to the perimeter. This equation holds for other gyro configurations, triangular, and circular fiber-optic gyros.

### 6.1.2 Fiber-Optic Gyros

The FOG is a maturing technology. It is similar to the RLG in that it is a single-degree-of-freedom sensor and its principle of operation is based on optical and electronic phenomena. This device is shown schematically in Fig. 6.4. Optical fiber is wound around a spool that is the closed optical path for the gyro. The Sagnac effect is also exploited in this gyro's mechanization.

The interferometric FOG's phase shift output can be determined from the frequency shift given in Eq. (6.13). The phase shift is related to frequency shift by

$$\Delta\phi = \frac{2\pi}{c} \Delta\nu NL \quad (6.14)$$

where  $N$  is the number of windings around spool and  $L$  is the length of optical fiber.

Substituting from Eq. (6.13), the phase shift becomes

$$\Delta\phi = \frac{2\pi(\pi D^2/4)4}{c \pi DN} \frac{NL}{\lambda}$$

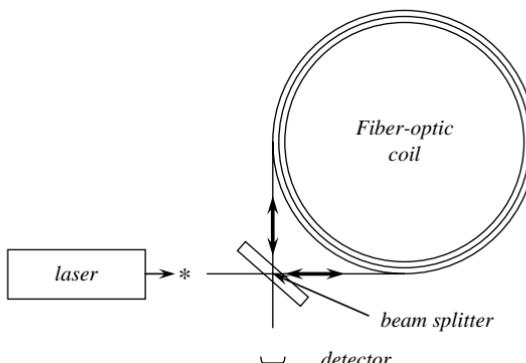


Fig. 6.4 Fiber-optic gyro.

or

$$\Delta\phi = \frac{2\pi LD}{c\lambda} \omega \quad (6.15)$$

This result shows that the phase change as a result of the gyro's rotation is more sensitive with the longer length fiber.

## 6.2 Sensor Error Models

Inertial sensor component (instrument) errors create error in the navigation system's computed position, velocity, and attitude, as indicated in the linearized equations developed in Chapter 5. These instrument errors can be represented in a general form, including some significant environment-dependent errors, as shown in Fig. 6.5 for accelerometers and in Fig. 6.6 for gyros. These errors are expressed in a "case" frame, which is usually the platform frame for gimballed inertial navigation units and the body frame for strap-down inertial navigation units.

The errors presented in these figures do not include other environment-sensitive terms, for example, gyro *g*-sensitive drift, scale-factor asymmetry, etc. The specific form for these errors can depend on the design of the instrument, and therefore, they are specialized for a specific design.

### 6.2.1 Random Constant Errors

Both the accelerometer and gyro models contain time-correlated random errors. The other terms, for example, biases and scale-factor/misalignments,

$$\begin{aligned}
 \begin{bmatrix} \delta f_x \\ \delta f_y \\ \delta f_z \end{bmatrix} &= \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} && \text{Bias -long term stable error} \\
 &+ \begin{bmatrix} aSF_x & aMA_{xy} & aMA_{xz} \\ aMA_{yx} & aSF_y & aMA_{yz} \\ aMA_{zx} & aMA_{zy} & aSF_z \end{bmatrix} \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} && \text{Scale-Factor and MisAlignments} \\
 &+ [\dots \dots \dots] && \text{Other environment-sensitive errors:- scale factor asymmetry} \\
 &&& \text{-g-squared sensitivity} \\
 &+ \begin{bmatrix} a_2 B_x \\ a_2 B_y \\ a_2 B_z \end{bmatrix} && \text{Short-term error (time correlated)} \\
 &+ \begin{bmatrix} aW_x \\ aW_y \\ aW_z \end{bmatrix} && \text{White noise}
 \end{aligned}$$

Fig. 6.5 General accelerometer error model.

$$\begin{aligned}
 \begin{bmatrix} \mathcal{E}_x \\ \mathcal{E}_y \\ \mathcal{E}_z \end{bmatrix} &= \begin{bmatrix} gB_x \\ gB_y \\ gB_z \end{bmatrix} && \text{Bias -long term stable error} \\
 &+ \begin{bmatrix} gSF_x & gMA_{xy} & gMA_{xz} \\ gMA_{yx} & gSF_y & gMA_{yz} \\ gMA_{zx} & gMA_{zy} & gSF_z \end{bmatrix} \begin{bmatrix} \omega_{i/bx}^b \\ \omega_{i/by}^b \\ \omega_{i/bz}^b \end{bmatrix} && \text{Scale-Factor and MisAlignments} \\
 &+ [\dots \quad \dots \quad \dots] && \text{Other environment-sensitive errors:-scale factor asymmetry} \\
 &+ \begin{bmatrix} g_s B_x \\ g_s B_y \\ g_s B_z \end{bmatrix} && \text{Short-term error (time correlated)} \\
 &+ \begin{bmatrix} g W_x \\ g W_y \\ g W_z \end{bmatrix} && \text{White noise}
 \end{aligned}$$

**Fig. 6.6 General gyro model.**

can be considered random constants whose dynamic model is simply

$$\dot{c} = 0 \quad (6.16)$$

Models for time-correlated error terms are presented next.

### 6.2.2 Random Process Dynamic Models

In the following, discrete model forms suitable for computer modeling and simulation of time-correlated and random walk models are developed. The statistical properties of random processes are also developed. These properties are useful in evaluating the computer model's outputs to ensure that the model implements the process correctly.

*Continuous time: Continuous systems approach.* In this section, dynamic models are assumed to be given in a continuous-time form. Consider the following scalar random variable, modeled as a continuous process, whose differential equation is

$$\dot{x}(t) = -\beta x(t) + w(t) \quad (6.17)$$

The solution of this equation is

$$x(t) = \exp[-\beta(t - t_0)]x(t_0) + \int_{t_0}^t \exp[-\beta(t - \lambda)]w(\lambda) d\lambda \quad (6.18)$$

The initial condition and driving noise are assumed to be zero mean, that is,

$$E[x(t_0)] = E[w(\lambda)] = 0 \quad (6.19)$$

and the driving noise is assumed to be a white-noise process such that

$$E[w(t)w(\tau)] = N\delta(t - \tau) \quad (6.20)$$

and the initial value and driving noise are assumed to be uncorrelated such that

$$E[x(t_0)w(\tau)] = 0 \quad (6.21)$$

Statistical properties of this process are determined next. It can be readily shown that  $x(t)$ , based on the preceding assumptions, is a zero mean process:

$$E[x(t)] = e^{-\beta(t-t_0)}E[x(t_0)] + \int_{t_0}^t e^{-\beta(t-\lambda)}E[w(\lambda)]d\lambda \quad (6.22)$$

The variance of  $x(t)$  is determined from the square of  $x(t)$ :

$$\begin{aligned} x^2(t) &= \exp[-2\beta(t-t_0)]x^2(t_0) + 2\exp[-\beta(t-t_0)]x(t_0) \int_{t_0}^t \exp[-\beta(t-\lambda)]w(\lambda)d\lambda \\ &\quad \xleftarrow[1]{\longrightarrow} \quad \xleftarrow[2]{\longrightarrow} \quad \xrightarrow[3]{\longrightarrow} \\ &\quad + \int_{t_0}^t \exp[-\beta(t-\lambda)]w(\lambda)d\lambda \int_{t_0}^t \exp[-\beta(t-\mu)]w(\mu)d\mu \\ &\quad \xleftarrow[3]{\longrightarrow} \quad \xrightarrow[2]{\longrightarrow} \quad \end{aligned} \quad (6.23)$$

Examining each of the indicated terms in Eq. (6.23), corresponding variances are rewritten as follows:

$$1) \quad \exp[-2\beta(t-t_0)]E[x^2(t_0)]$$

$$2) \quad 2\exp[-\beta(t-t_0)] \int_{t_0}^t \exp[-\beta(t-\lambda)]E[x(t_0)w(\lambda)]d\lambda = 0$$

$$\begin{aligned} 3) \quad E\left[\int_{t_0}^t \exp[-\beta(t-\lambda)]w(\lambda)d\lambda \int_{t_0}^t \exp[-\beta(t-\mu)]w(\mu)d\mu\right] \\ &= \int_{t_0}^t \exp[-\beta(t-\lambda)] \int_{t_0}^t \exp[-\beta(t-\mu)]E[w(\lambda)w(\mu)]d\mu d\lambda \\ &= \int_{t_0}^t \exp[-\beta(t-\lambda)] \int_{t_0}^t \exp[-\beta(t-\mu)]N\delta(t-\mu)d\mu d\lambda \\ &= N\exp(-2\beta t) \int_{t_0}^t \int_{t_0}^t \exp[-\beta(\lambda+\mu)]\delta(t-\mu)d\mu d\lambda \\ &= N\exp(-2\beta t) \int_{t_0}^t \exp[-2\beta\lambda]d\lambda \\ &= \frac{N}{2\beta}[1 - \exp[-2\beta(t-t_0)]] \end{aligned}$$

Or, summarizing this result,

$$E[x^2(t)] = \exp[-2\beta(t - t_0)]E[x^2(t_0)] + \frac{N}{2\beta}[1 - \exp[-2\beta(t - t_0)]] \quad (6.24)$$

The steady-state variance of this equation is evaluated in the following, redefining the times  $t$  and  $t_0$  as

$$\begin{aligned} t &\leftarrow t + \tau \\ t_0 &\leftarrow t \end{aligned}$$

The expected value of  $x^2(t)$  at time  $t + \tau$  becomes

$$E[x^2(t + \tau)] = e^{-2\beta\tau}E[x^2(t)] + \frac{N}{2\beta}[1 - e^{-2\beta\tau}] \quad (6.25)$$

Steady state is defined when

$$E[x^2(t + \tau)] = E[x^2(t)] = \sigma^2$$

where

$$\sigma^2 = \frac{N}{2\beta} \quad (6.26)$$

The specified steady-state value of  $E[x^2(t)]$  is determined by the value of  $N$ .

A *discrete-time* form of the continuous process is developed next. Equation (6.18) is expressed as the following discrete process:

$$x_{k+1} = e^{-\beta\Delta t}x_k + w_{k+1} \quad (6.27)$$

where

$$w_{k+1} = \int_{t_k}^{t_{k+1}} \exp[-\beta(t - \lambda)]w(\lambda) d\lambda \quad (6.28)$$

and

$$\Delta t = t_{k+1} - t_k \quad \forall \quad t_k$$

The variance of  $x_{k+1}$  is determined by first writing out several terms of Eq. (6.27):  
 $k = 0$ :

$$x_1 = e^{-\beta\Delta t}x_0 + w_1$$

$k = 1$ :

$$\begin{aligned}x_2 &= e^{-\beta\Delta t}x_1 + w_2 \\&= e^{-2\beta\Delta t}x_0 + e^{-\beta\Delta t}w_1 + w_2\end{aligned}$$

$k = 2$ :

$$\begin{aligned}x_3 &= e^{-\beta\Delta t}x_2 + w_3 \\&= e^{-3\beta\Delta t}x_0 + e^{-2\beta\Delta t}w_1 + e^{-\beta\Delta t}w_2 + w_3\end{aligned}$$

The accumulated sum at time  $k+1$  becomes

$$\begin{aligned}x_{k+1} &= \exp[-\beta(k+1)\Delta t]x_0 + \exp[-\beta(k)\Delta t]w_1 \\&\quad + \exp[-\beta(k-1)\Delta t]w_2 + \cdots + \exp(-\beta\Delta t)w_k + w_{k+1}\end{aligned}\quad (6.29)$$

Assuming, as before, that the initial condition  $x_0$  and the random sequence  $w_k$  are independent, and that variances  $E[w_i^2]$  are equal and equal to  $\nu^2$ , the variance of Eq. (6.29) becomes

$$\begin{aligned}E[x_{k+1}^2] &= \exp[-2\beta(k+1)\Delta t]E[x_0^2] + \nu^2[1 + \exp(-2\beta\Delta t) \\&\quad + \cdots + \exp[-2\beta(k-1)\Delta t] + \exp[-2\beta(k)\Delta t]] \\&= \exp[-2\beta(k+1)\Delta t]E[x_0^2] + \nu^2 \frac{[1 - \exp[-2\beta(k+1)\Delta t]]}{[1 - \exp(-2\beta\Delta t)]}\end{aligned}\quad (6.30)$$

Equation (6.24) can be rewritten in discrete form as

$$E[x_{k+1}^2] = \exp[-2\beta(k+1)\Delta t]E[x_0^2] + \frac{\nu^2}{2\beta}[1 - \exp[-2\beta(k+1)\Delta t]]\quad (6.31)$$

Because Eqs. (6.30) and (6.31) are equivalent, the following relationship between continuous and discrete driving noise characteristics is obtained:

$$\begin{aligned}\nu^2 &= \frac{N}{2\beta}[1 - e^{-2\beta\Delta t}] \\&= \sigma^2[1 - e^{-2\beta\Delta t}]\end{aligned}\quad (6.32)$$

The discrete form of this process, expressed in terms of the continuous process's steady-state variance  $\sigma^2$ , becomes

$$\begin{aligned}x_{k+1} &= e^{-\beta\Delta t}x_k + w_{k+1} \\&= e^{-\beta\Delta t}x_k + \nu u_{k+1} \\&= e^{-\beta\Delta t}x_k + \{\sigma^2[1 - e^{-2\beta\Delta t}]\}^{1/2}u_{k+1}\end{aligned}\quad (6.33)$$

where  $u_i$  = unit variance samples from a Gaussian random number generator. Assume that the following approximation is true:

$$e^{-\beta\Delta t} \approx 1 - \beta\Delta t \quad (6.34)$$

Then, Eq. (6.33) can be rewritten as

$$\begin{aligned} x_{k+1} &= (1 - \beta\Delta t)x_k + \{\sigma^2[1 - e^{-2\beta\Delta t}]\}^{1/2}u_{k+1} \\ &= x_k - \beta x_k \Delta t + \sigma(2\beta\Delta t)^{1/2}u_{k+1} \end{aligned} \quad (6.35)$$

Defining the differential

$$\Delta x_{k+1} = x_{k+1} - x_k \quad (6.36)$$

then the following stochastic differential equation form is obtained for the time-correlated process:

$$\Delta x_{k+1} = -\beta x_k \Delta t + \sigma(2\beta)^{1/2}u_{k+1}\sqrt{\Delta t} \quad (6.37)$$

As the preceding approximation suggests,  $\beta$  being small, Eq. (6.37) describes a random walk process. Using Eq. (6.26) in Eq. (6.37) yields

$$\begin{aligned} \Delta x_{k+1} &= -\beta x_k \Delta t + \sigma \frac{\sqrt{N}}{\sigma} u_{k+1} \sqrt{\Delta t} \\ &= -\beta x_k \Delta t + \sqrt{N}u_{k+1}\sqrt{\Delta t} \end{aligned} \quad (6.38)$$

which, after setting  $\beta$  to zero, yields the following for the random walk process:

$$\Delta x_{k+1}(\beta = 0) = \sqrt{N}u_{k+1}\sqrt{\Delta t} \quad (6.39)$$

Next, these results will be shown to be the same as that obtained by using a stochastic differential equation approach.

*Stochastic differential equation approach.* Following Astrom [9], consider the following vector stochastic differential equation:

$$dx(t) = A(t)x(t)dt + d\mu(t) \quad (6.40)$$

where  $d\mu(t)$  is a Brownian motion (independent increment) process. In a continuous form, Eq. (6.40) is often written as

$$\dot{x}(t) = A(t)x(t) + w(t) \quad (6.41)$$

which implies

$$w(t) = \frac{d\mu(t)}{dt} \quad (6.42)$$

The derivative in Eq. (6.42) represents white noise; however, the continuous Gaussian process  $w(t)$  is not differentiable.

The covariance of Eq. (6.40) is found from the following:

$$\begin{aligned} dP(t) &= E\{[\mathbf{x}(t) + d\mathbf{x}(t)][\mathbf{x}(t) + d\mathbf{x}(t)]^T - \mathbf{x}(t)\mathbf{x}^T(t)\} \\ &= \{E[\mathbf{x}(t)\mathbf{x}^T(t)]A^T(t) + A(t)E[\mathbf{x}(t)\mathbf{x}^T(t)]\}dt + E[d\boldsymbol{\mu} d\boldsymbol{\mu}^T] + \mathcal{O}(dt) \\ &= [P(t)A^T(t) + A(t)P(t) + Q(t)]dt \end{aligned} \quad (6.43)$$

This result is obtained assuming that the instantaneous state value  $\mathbf{x}(t)$  and  $d\boldsymbol{\mu}(t)$  are uncorrelated and  $Q(t)$  is defined by

$$E[d\boldsymbol{\mu} d\boldsymbol{\mu}^T] \equiv Q(t) dt \quad (6.44)$$

Equation (6.44) implies

$$d\boldsymbol{\mu}(t) \approx \mathbf{u}(t)\sqrt{dt} \quad (6.45)$$

where

$$Q(t) = E[\mathbf{u}(t)\mathbf{u}(t)^T] \quad (6.46)$$

Therefore, Eq. (6.40) can be expressed as

$$d\mathbf{x}(t) = A(t)\mathbf{x}(t) dt + \mathbf{u}(t)\sqrt{dt} \quad (6.47)$$

Equation (6.47) will be specialized for the two scalar processes, random walk and first-order correlated process, in the following.

For a random walk process  $A(t)$  is zero. The random walk can be expressed by the following scalar stochastic differential equation:

$$dx_{RW}(t) = u_{RW}(t)\sqrt{dt} \quad (6.48)$$

Or, if implemented as a time derivative in a simulation, then

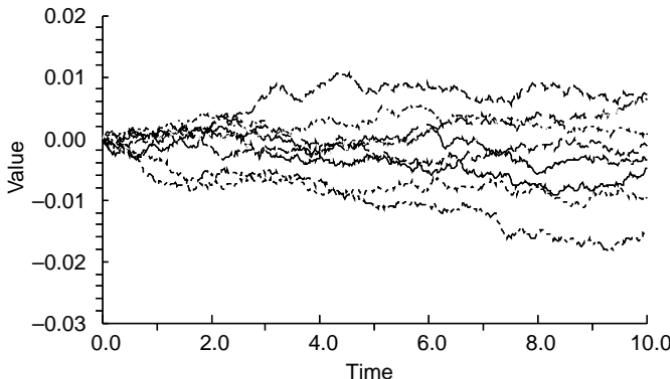
$$\dot{x}_{RW}(t) = \frac{u_{RW}(t)}{\sqrt{dt}} \quad (6.49)$$

The covariance of this process becomes

$$dp_{RW}(t) = q_{RW}(t) dt \quad (6.50)$$

or

$$\dot{p}_{RW}(t) = q_{RW}(t) \quad (6.51)$$

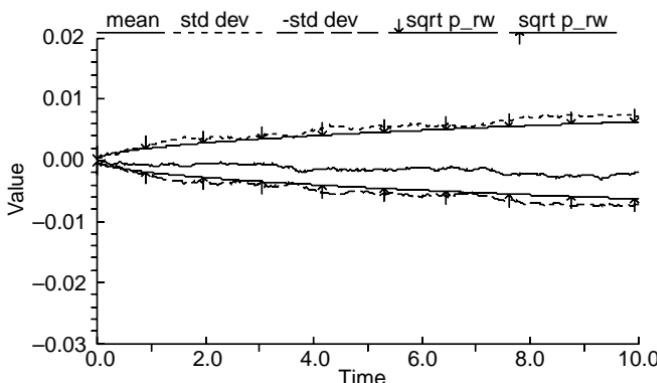


**Fig. 6.7** Monte Carlo simulation of random walk.

If, for example, this process is the model for the RLG random drift,  $u_{RW}$  is specified in angle error per root time (deg/ $\sqrt{h}$ ), and then the differential covariance  $dp_{RW}$  has units of angle squared.

Monte Carlo simulation results implementing Eq. (6.48) are presented in Fig. 6.7. Shown are several sample paths (time histories) of the random walk process. Shown in Fig. 6.8 are the corresponding computed standard deviations (one-sigma values) and the results of integrating Eq. (6.51). Standard deviations and variances in Fig. 6.8 are approximately the same. As indicated by Fig. 6.8 and expressed in Eq. (6.51), the variance for the random walk continues to increase with time.

A time-correlated process is specified by the process time constant  $\tau$  and the driving noise  $w(t)$  in Eq. (6.41). It is desirable to have the variance of stationary process reach a known steady-state value by specifying the magnitude of the process noise  $Q$  ( $P_{ss} = Q$ ). This is achieved by the following time-invariant



**Fig. 6.8** Standard deviation and variance for random walk.

scalar stochastic differential equation:

$$dx_c = -\frac{1}{\tau}x_c dt + \frac{\sqrt{2}}{\sqrt{\tau}} w_c \sqrt{dt} \quad (6.52)$$

This equation is identical to Eq. (6.37) if  $w_c$  is replaced with  $\sigma$  and  $1/\tau$  is replaced by  $\beta$ . If Eq. (6.52) is implemented at the time-derivative level in a simulation, then

$$\dot{x}_c = -\frac{1}{\tau}x_c + \frac{\sqrt{2}}{\sqrt{\tau}} w_c / \sqrt{dt} \quad (6.53)$$

The corresponding variance differential equation is

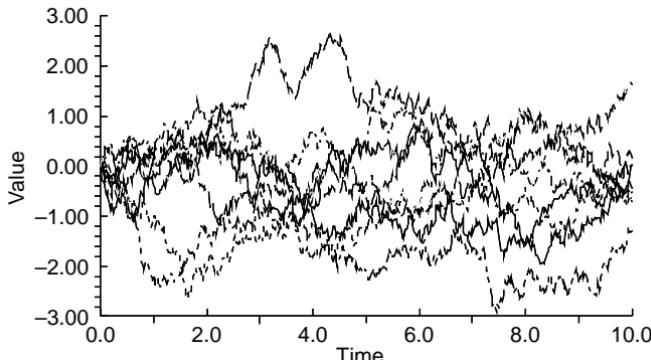
$$\dot{p}_c = -\frac{2}{\tau}p_c + \frac{2}{\tau}q_c \quad (6.54)$$

The steady-state condition,  $\dot{p}_c = 0$ , for this process is

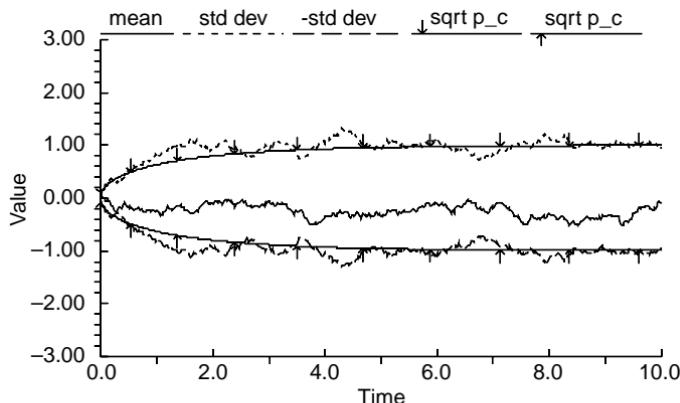
$$p_{ss} \equiv q_c \quad (6.55)$$

Monte Carlo simulation results implementing Eq. (6.52) are presented in Fig. 6.9. Shown are several sample paths (time histories) of the time-correlated process. Figure 6.10 shows the corresponding computed standard deviations (one-sigma values) and the results of integrating Eq. (6.54). Standard deviations and variance results in Fig. 6.10 agree. The steady-state condition described by Eq. (6.55) can be seen in Fig. 6.10.

Simulation results presented here and those presented for random walk exhibit a small nonzero mean. This is an artifice of the simulation's uniform random number generator. Caution should be exercised in using results such as these



**Fig. 6.9 Monte Carlo simulation of correlated error.**



**Fig. 6.10 Standard deviation and variance for correlated error.**

in evaluating a filter's estimates of similarly modeled inertial instrument errors, that is, accelerometer and gyro random errors. This nonzero mean characteristic can be incorrectly estimated by a filter as one of these instrument's bias values.

### 6.3 Chapter Summary

The Sagnac effect that governs the operation of both RLGs and FOGs was described at a simplified level. Simple performance characterizations of these sensors were presented to illustrate how the gyro's design, that is, RLG path length, influences the gyro's performance.

General mathematical models representing gyro and accelerometer errors were presented, including long-term stable errors and short-term random errors. Dynamic models for time-correlated random processes and random walk errors were developed, and examples of these errors' time-varying and statistical characteristics were illustrated using numerical simulations of these processes.

Selected elements of these models are included within the error state structure describing the navigation system's errors. The resulting system error model is implemented in the Kalman filter algorithm (see Sec. 11.2). The gyro and accelerometer error models are used in a case study examining the calibration of these sensors in a laboratory (see Chapter 9).

### Problems

- 6.1** This exercise presents an illustration of including sensor errors as additional error states in a system error dynamics matrix.

Consider the body-referenced instrument (accelerometer or gyro) errors

$$\begin{bmatrix} \underline{\delta}_x \\ \underline{\delta}_y \\ \underline{\delta}_z \end{bmatrix}^b = \begin{bmatrix} -B_x \\ -B_y \\ -B_z \end{bmatrix} + \begin{bmatrix} -SF_x & -MA_{xy} & -MA_{xz} \\ -MA_{yx} & -SF_y & -MA_{zx} \\ -MA_{zx} & -MA_{zy} & -SF_z \end{bmatrix} \begin{bmatrix} -x \\ -y \\ -z \end{bmatrix}^b + \dots$$

The preceding underline “ ” can be replaced by accelerometer (specific force) or gyro symbols as in Figs. 6.5 and 6.6. The environmental stimulation for the scale-factor and misalignment errors  $\begin{bmatrix} -x \\ -y \\ -z \end{bmatrix}^b$  are body-referenced specific forces or angular rates. This body-referenced error is transformed to the navigation frame for the filter dynamics model as

$$\begin{bmatrix} \underline{\delta}_x \\ \underline{\delta}_y \\ \underline{\delta}_z \end{bmatrix}^n = C_b^n \begin{bmatrix} \underline{\delta}_x \\ \underline{\delta}_y \\ \underline{\delta}_z \end{bmatrix}^b$$

It is desirable to estimate the body-referenced errors by including these in the filter error state vector, because, for strap-down navigation systems, only the body-referenced sensor errors are relatively “constant” and lend themselves to a dynamic model, modeling these errors as random constants. The state vector partition, corresponding to these errors, is written as (transposed)

$$\begin{bmatrix} \cdot & -B_x & -B_y & -B_z & -SF_x & -MA_{xy} & -MA_{xz} & -MA_{yx} \\ & -SF_y & -MA_{yz} & -MA_{zx} & -MA_{zy} & SF_z & \cdot \end{bmatrix}$$

The transformation of these errors from the body frame to the navigation frame is accomplished by the following expression. This form for the relevant errors is inserted into the velocity error states for accelerometer errors and into the attitude error states for gyro errors.

The following result is obtained from a sequence of steps.

$$\begin{bmatrix} \underline{\delta}_x \\ \underline{\delta}_y \\ \underline{\delta}_z \end{bmatrix}^n = \begin{bmatrix} C_{xx}^{-x} & C_{xx}^{-y} & C_{xx}^{-z} & C_{xy}^{-x} & C_{xy}^{-y} & C_{xy}^{-z} & C_{xz}^{-x} & C_{xz}^{-y} & C_{xz}^{-z} \\ C_b^n | C_{yx}^{-x} & C_{yx}^{-y} & C_{yx}^{-z} & C_{yy}^{-x} & C_{yy}^{-y} & C_{yy}^{-z} & C_{yz}^{-x} & C_{yz}^{-y} & C_{yz}^{-z} \\ C_{zx}^{-x} & C_{zx}^{-y} & C_{zx}^{-z} & C_{zy}^{-x} & C_{zy}^{-y} & C_{zy}^{-z} & C_{zz}^{-x} & C_{zz}^{-y} & C_{zz}^{-z} \end{bmatrix} \begin{bmatrix} \underline{\mathbf{B}} \\ -SF_x \\ -MA_{xy} \\ -MA_{xz} \\ -MA_{yx} \\ -SF_y \\ -MA_{yz} \\ -MA_{zx} \\ -MA_{zy} \\ -SF_z \end{bmatrix}$$

First, expand the error model to separate the environment-sensitive errors from the bias errors:

$$\begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}^n = C_b^n \begin{bmatrix} -B_x \\ -B_y \\ -B_z \end{bmatrix} + C_b^n \begin{bmatrix} -SF_x & -MA_{xy} & -MA_{xz} \\ -MA_{yx} & -SF_y & -MA_{yz} \\ -MA_{zx} & -MA_{zy} & -SF_z \end{bmatrix} \begin{bmatrix} -x \\ -y \\ -z \end{bmatrix}^b$$

Next, multiply through by the environment stimuli

$$\begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}^n = C_b^n \begin{bmatrix} -B_x \\ -B_y \\ -B_z \end{bmatrix} + C_b^n \begin{bmatrix} -SF_x -x + -MA_{xy} -y + -MA_{xz} -z \\ -MA_{yx} -x + -SF_y -y + -MA_{yz} -z \\ -MA_{zx} -x + -MA_{zy} -y + -SF_z -z \end{bmatrix}$$

Define an error state vector composed of the sensor errors and configure the multiplying terms to reproduce the preceding result:

$$\begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}^n = C_b^n \begin{bmatrix} -B_x \\ -B_y \\ -B_z \end{bmatrix} + C_b^n \begin{bmatrix} -x & -y & -z & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -x & -y & -z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -x & -y & -z \end{bmatrix} \times \begin{bmatrix} -SF_x \\ -MA_{xy} \\ -MA_{xz} \\ -MA_{yx} \\ -SF_y \\ -MA_{yz} \\ -MA_{zx} \\ -MA_{zy} \\ -SF_z \end{bmatrix}$$

Finally, multiply through by the  $C_b^n$  matrix to obtain the preceding result.

- 6.2** This exercise concerns the dual-axis sensor error model. Error models for dual-axis sensors contain common errors that are the result of common effects for the two axes. For example, a tuned rotor gyro has common acceleration and rate sensitive errors in the axes normal to the spinning rotor shaft. Consider the following gyro error model for a dual-axis sensor whose spin axis is parallel to its  $z$  axis:

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ - \end{bmatrix}_z = \begin{bmatrix} gB_x \\ gB_y \\ gB_z \end{bmatrix}_z + \begin{bmatrix} gSF_x & gMA_{xy} & -gMA_{xz} \\ -gMA_{xy} & gSF_x & gMA_{yz} \\ gMA_{xz} & -gMA_{yz} & sSF_z \end{bmatrix}_z \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \begin{bmatrix} gGS_x & gGS_{xy} & 0 \\ -gGS_{xy} & gGS_x & 0 \\ 0 & 0 & 0 \end{bmatrix}_z \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix}$$

where the gyro misalignments,  $gMA$ , result from small-angle approximations to the transformation matrix transforming from the sensor's case to the navigation system's reference axes.

Rearrange this dual-axis model into the following form:

$$\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ - \end{bmatrix}_z = \begin{bmatrix} 1 & 0 & 0 & \omega_x & 0 & \omega_y & -\omega_z & 0 & f_x & f_y \\ 0 & 1 & 0 & \omega_y & 0 & -\omega_x & 0 & \omega_z & f_y & -f_x \\ 0 & 0 & 1 & 0 & \omega_z & 0 & \omega_x & -\omega_y & 0 & 0 \end{bmatrix} \begin{bmatrix} gB_x \\ gB_y \\ gB_z \\ gSF_x \\ gSF_z \\ gMA_{xy} \\ gMA_{xz} \\ gMA_{yz} \\ gGS_x \\ gGS_{xy} \end{bmatrix}_z$$

- 6.3** This exercise concerns redundant dual-axis sensors' error model. If an additional dual-axis sensor is used, one of its axes is redundant. Consider an error model with the sensor's spin axis parallel to its  $x$  axis, and show that the combined error model for the two dual-axis sensors becomes

$$\begin{bmatrix} \varepsilon_{x,z} \\ \varepsilon_{y,z} \\ \varepsilon_{z,x} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \omega_x & 0 & \omega_y & -\omega_z & 0 & 0 & 0 & f_x & f_y & 0 & 0 \\ 0 & 1 & 0 & \omega_y & 0 & -\omega_x & 0 & \omega_z & 0 & 0 & f_y & -f_x & 0 & 0 \\ 0 & 0 & 1 & 0 & \omega_z & 0 & 0 & 0 & \omega_x & -\omega_y & 0 & 0 & f_z & -f_y \end{bmatrix} \begin{bmatrix} gB_{x,z} \\ gB_{y,z} \\ gB_{z,x} \\ gSF_{x,z} \\ gSF_{z,x} \\ gMA_{xy,z} \\ gMA_{xz,z} \\ gMA_{yz,z} \\ gMA_{xz,x} \\ gMA_{xy,x} \\ gGS_{x,z} \\ gGS_{xy,z} \\ gGS_{y,x} \\ gGS_{yx,x} \end{bmatrix}_z$$

where the “,( )” subscript denotes the corresponding sensor's spin axis direction.

## Navigation Aids

Descriptions of aids to navigation presented in this chapter include the following: Doppler velocity sensors (DVS), tactical-air-navigation (TACAN) range, global positioning systems (GPS), and forward-looking infrared (FLIR) and other line-of-sight (LOS) systems. These systems provide independent redundant navigation information that can aid an integrated navigation system via the Kalman filter algorithm.

Doppler velocity sensors are used widely in aircraft, seacraft, and land vehicles. While operating with different mechanizations, for example, radar, acoustic, laser, etc., most employ the Doppler shift principle. A simple characterization of a Doppler system's operation, performance characteristics, and development of error modeling approaches, assuming the navigation system error model representations defined in Chapter 5, is presented.

GPS operates on a satellite-to-receiver/antenna ranging principle. This is functionally similar to the principle of operation for TACAN. TACAN ranging is used with respect to a fixed ground station, whereas GPS ranging is used with respect to a satellite whose position is known relatively accurately. In this presentation, TACAN is used to illustrate a single ranging operation that is extended to multiple-satellite ranges for GPS.

Presented are Kepler's equations that describe satellite orbital motion and are used in the GPS system's processing to determine the satellite positions in an Earth-centered, Earth-fixed (ECEF) frame. Quality measures associated with GPS data are used to determine which satellite's ranging data are to be used. The satellite elevation and geometrical-dilution-of-precision (GDOP) quality measures are developed.

Airborne line-of-sight systems include radar, FLIR, television (TV), etc. These systems serve many functions, and they can provide aiding information to a navigation system. Error models are developed, using navigation system error models in Chapter 5, for incorporating this system's data to a Kalman filter algorithm. Because the airborne LOS systems are closely related, only the FLIR system is presented.

Problems are included to expand on the material presented.

## 7.1 Doppler Velocity Sensors

### 7.1.1 Doppler Velocity Sensor Functionality

The Doppler effect can be explained by first considering a stationary emitting source that emits waves with period  $\tau$ . The number of oscillations meeting a stationary observer over a time interval  $\Delta t$  is

$$N = \frac{\Delta t}{\tau} \quad (7.1)$$

If the observer moves toward the stationary source with velocity  $V$ , the number of additional oscillations  $\Delta N$  is (stationary source with moving observer)

$$\Delta N = \frac{V \Delta t}{\lambda} \quad (7.2)$$

and, if the source is emitted from the observer and reflected back toward the observer, then these additional oscillations are doubled as (reflected moving source at observer)

$$\Delta N = \frac{2V \Delta t}{\lambda} \quad (7.3)$$

The frequency change is the number of additional oscillations during the time interval

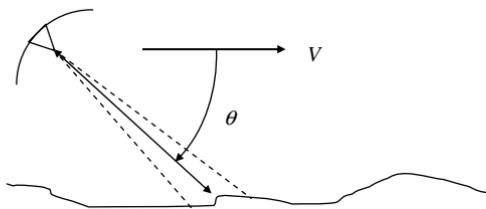
$$\begin{aligned} \Delta\nu &= \frac{\Delta N}{\Delta t} \\ &= \frac{2V}{\lambda} \end{aligned} \quad (7.4)$$

The equation describing the Doppler shift becomes

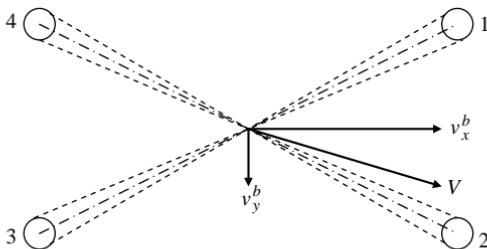
$$\Delta\nu = \frac{2\nu V}{c} \quad (7.5)$$

If the emitter/observer's velocity is not parallel to the reflecting source but is at a slant angle  $\theta$ , as shown in Fig. 7.1, then the frequency change is modified by

$$\Delta\nu = \frac{2\nu V}{c} \cos \theta \quad (7.6)$$



**Fig. 7.1** Doppler at angle with respect to ground surface.



**Fig. 7.2 Janus Doppler configuration.**

The frequency change is a function of the velocity relative to the reflecting surface.

A commonly used Doppler mechanization is the “Janus” configuration, which uses four beams, two directed forward and two directed rearward, depressed downward and canted left and right, as shown in Fig. 7.2. Using the frequency changes from the four beams, velocity components referenced to the aircraft body axis can be formed as

$$V_x \propto \Delta\nu_1 + \Delta\nu_2 - \Delta\nu_3 - \Delta\nu_4 \quad (7.7)$$

$$V_y \propto \Delta\nu_2 + \Delta\nu_3 - \Delta\nu_1 - \Delta\nu_4 \quad (7.8)$$

and

$$V_z \propto \Delta\nu_1 + \Delta\nu_2 + \Delta\nu_3 + \Delta\nu_4 \quad (7.9)$$

The mechanization shown in Fig. 7.2 is relatively insensitive to terrain variations.

Doppler systems exhibit sensitivity while operating over water as a result of changes in surface reflectivity. This is dependent on the sea’s Beaufort state for over-ocean operations. Operations over a moving surface, such as water, corrupt the Doppler sensor’s measurements of velocity.

### 7.1.2 *Doppler Velocity Sensor Outputs for Navigation Filter Implementation*

The Doppler unit is assumed to be a body-fixed unit. (There are some Doppler units that are stabilized and isolated from the vehicle’s motion via a mechanical gimbal arrangement; to analyze these units, the knowledge of the gimbal configuration is required.) The Doppler velocity errors include the unit’s inherent errors, for example, scale factor, bias and noise, and errors resulting from the unit’s installation, that is, mechanical misalignment (bore-sight).

Shown in Fig. 7.3 is a generalized error model for the DVS. Bias and scale-factor/bore-sight errors can be considered as random constants. For Doppler units based on a radar principle, the noise can be a function of the vehicle’s velocity and is usually referred to as fluctuation noise.

$$\begin{aligned} \begin{bmatrix} \delta v_x^b_{\text{DOP}} \\ \delta v_y^b_{\text{DOP}} \\ \delta v_z^b_{\text{DOP}} \end{bmatrix} &= \begin{bmatrix} dB_x \\ dB_y \\ dB_z \end{bmatrix} && \text{Bias - long term stable error} \\ &+ \begin{bmatrix} dSF_x & dB S_{xy} & dB S_{xz} \\ dB S_{yx} & dSF_y & dB S_{yz} \\ dB S_{zx} & dB S_{zy} & dSF_z \end{bmatrix} \begin{bmatrix} v_x^b \\ v_y^b \\ v_z^b \end{bmatrix} && \text{Scale-factor and misalignments} \\ &+ \begin{bmatrix} dW_x \\ dW_y \\ dW_z \end{bmatrix} && \text{White noise} \end{aligned}$$

**Fig. 7.3 General Doppler velocity error model.**

The off-axis, velocity-dependent terms in this model represent misalignment or bore-sight errors. The  $dB S_{yz}$  and  $dB S_{zx}$  terms represent azimuth and pitch bore-sight errors, respectively. For fixed-wing aircraft applications, products of Doppler errors with the aircraft velocity components  $v_y^b$  and  $v_z^b$  are usually small enough to neglect relative to the Doppler's bias terms. However, for helicopters, the products with  $v_y^b$  might not be small enough to neglect because helicopters can develop significant lateral velocities.

Outputs from the DVS are combined with inertial navigation system velocities to form a measurement for processing in an onboard Kalman filter algorithm. The difference between the Doppler velocity and the inertial navigation system velocity is termed the Doppler divergence and is used as the observation in the Kalman filter. This is expressed as

$$\Delta v_{\text{DOP}} = v_{\text{INU}} - v_{\text{DOP}} \quad (7.10)$$

In Example 7.1, the specialization of this difference to the inertial navigation system perturbation error model presented in Chapter 5 will be presented.

### Example 7.1 Perturbation Velocity Error Representation

Expressing Eq. (7.10) in the navigation frame  $n$  and expanding about a nominal velocity  $v^n$  and neglecting products of error quantities yield

$$\begin{aligned} \Delta v_{\text{DOP}}^n &= \bar{v}_{\text{INU}}^n - \bar{v}_{\text{DOP}}^n \\ &= v_{\text{INU}}^n + \delta v_{\text{INU}}^n - \bar{C}_b^n (v_{\text{DOP}}^b + \delta v_{\text{DOP}}^b) \\ &= v_{\text{INU}}^n + \delta v_{\text{INU}}^n - [I - (\boldsymbol{\phi} \times)] C_b^n (v_{\text{DOP}}^b + \delta v_{\text{DOP}}^b) \\ &\approx v_{\text{INU}}^n - C_b^n v_{\text{DOP}}^b + \delta v_{\text{INU}}^n - C_b^n \delta v_{\text{DOP}}^b + (\boldsymbol{\phi} \times) C_b^n v_{\text{DOP}}^b \\ &= \delta v_{\text{INU}}^n + (\boldsymbol{\phi} \times) C_b^n v_{\text{DOP}}^b - C_b^n \delta v_{\text{DOP}}^b \\ &= \delta v_{\text{INU}}^n - (v^n \times) \boldsymbol{\phi} - C_b^n \delta v_{\text{DOP}}^b \end{aligned}$$

The first terms corresponding to the navigation system and then the Doppler error terms of the linearized measurement matrix become

$$\begin{aligned} H_{DOP} &= \frac{\partial \Delta v}{\partial \delta \theta, \delta v, \phi, \dots, \delta v_{DOP}^b} \\ &= [0_{3 \times 3} \quad I_{3 \times 3} \quad -(v^n \times) \cdots -C_b^n] \end{aligned}$$

## 7.2 Tactical-Air-Navigation Range

### 7.2.1 Tactical-Air-Navigation Functionality

TACAN provides a range measurement from the unit's position to a fixed ground station, as illustrated in Fig. 7.4.

Other data provided by some TACAN units include magnetic bearing and range rate; however, these data might not be of sufficient quality for use as navigation aids. Magnetic bearing is corrupted by local electromagnetic fields, and range rate is a derived value. Some TACAN units provide two channels for two sets of range, bearing and range rate.

### 7.2.2 Tactical-Air-Navigation Range for Navigation Filter Implementation

TACAN range is the magnitude of the relative position vector and is formed next by the sum of the squares of these components:

$$\rho_{TACAN} = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} + b_{TACAN} + v_{TACAN} \quad (7.11)$$

where corrupting bias and measurement noise have been added to the measurement model. The range observation is formed as the difference between this measured range and that computed based on the navigation systems' position

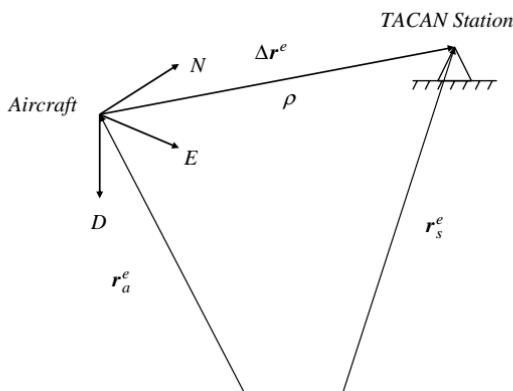


Fig. 7.4 TACAN ranging.

relative to the TACAN station's coordinates (see Problem 2.7). Linearizing this range difference with respect to the relative position vector elements and biases results in the following linearized measurement matrix:

$$H_{\Delta\rho} = \frac{\partial \Delta\rho}{\partial \delta r, \delta v, \dots, b} = \begin{bmatrix} \frac{\partial \Delta\rho}{\partial \Delta x} & \frac{\partial \Delta\rho}{\partial \Delta y} & \frac{\partial \Delta\rho}{\partial \Delta z} & \dots & 1 \end{bmatrix} \quad (7.12)$$

If Kalman filter state vector definitions consist of a linear position error as in Eq. (7.11), the result in Eq. (7.12) suffices; however, other filter state vector definitions consisting of a different description of position error, that is, angular position error, then the measurement matrix is formed using the chain differentiation rule as

$$\frac{\partial \Delta\rho}{\partial \delta \theta} = \frac{\partial \Delta\rho}{\partial \Delta r} \frac{\partial \Delta r}{\partial \delta \theta} \quad (7.13)$$

The first of these partial derivatives is given in Eq. (7.12). Different Kalman filter state vector definitions result in different measurement matrix contents.

The use of TACAN range is functionally similar to GPS, which will be presented in the next section. An example illustrating the integration of TACAN and an inertial navigation unit (INU) via a Kalman filter algorithm will be presented in Chapter 10. The following example illustrates the change of variable indicated in Eq. (7.13).

### Example 7.2 TACAN Range Measurement Matrix

The TACAN range measurement model is to be formulated. The relative position vector between the TACAN station and the aircraft position, formed using the inertial navigation system's outputs, is expressed as

$$\Delta r^e = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}^e = C_z^x \left\{ \begin{bmatrix} (R_n + h_s) \cos \phi_s \cos \lambda_s \\ (R_n + h_s) \cos \phi_s \sin \lambda_s \\ [R_n(1 - \varepsilon^2) + h_s] \sin \phi_s \end{bmatrix} - \begin{bmatrix} (R_n + h) \cos \bar{\phi} \cos \bar{\lambda} \\ (R_n + h) \cos \bar{\phi} \sin \bar{\lambda} \\ [R_n(1 - \varepsilon^2) + h] \sin \bar{\phi} \end{bmatrix} \right\}$$

where the matrix

$$C_z^x = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

transforms from the ECEF frame, with the Earth rotation vector parallel to the  $z$  axis  $-C_n^e(z)$ , to the ECEF frame, with the Earth rotation vector parallel to the  $x$  axis  $-C_n^e(x)$  (see Sec. 3.1). The overbar represents corrupted latitude and longitude as computed by the navigation system.

In the preceding relative position equation, latitude error in the normal radius of curvature  $R_n$  is not included. It was shown in Problem 4.1 that the sensitivity of this value is of order  $\epsilon^2$ . This error's contribution is considered to be second-order and is neglected. Also neglected are altitude error affects because at the longer TACAN range, a large variation in altitude would be required for a significant effect on range.

TACAN range observation is linearized with respect to angular position state vector elements as illustrated in Eq. (7.13) using the chain differential rule. The first of these partial derivatives is presented in Eq. (7.12). The second derivative is applicable to this example, as well as other measurements to be formed later in this chapter, and will be developed next.

The computed latitude and longitude provided by the navigation system is imperfect. Representing the errors in latitude and longitude are the following:

$$\begin{aligned}\bar{\phi} &= \phi + \delta\phi \\ \bar{\lambda} &= \lambda + \delta\lambda\end{aligned}$$

Substituting these expressions into the preceding relative position range equation, expanding the sums of these angles using appropriate trigonometric identities, and neglecting products of error angles result in the following (see expressions of latitude and longitude error defined in terms of wander angle  $\alpha$  and angular position errors  $\delta\theta$  developed in Example 3.3):

$$\begin{aligned}\Delta\Delta\mathbf{r}^e &= C_z^\alpha \left\{ \begin{bmatrix} a_s c \phi_s c \lambda_s \\ a_s c \phi_s s \lambda_s \\ b_s s \phi_s \end{bmatrix} - \begin{bmatrix} a [c \phi c \lambda - s \phi c \lambda (s \alpha \delta \theta_x + c \alpha \delta \theta_y) - s \lambda (c \alpha \delta \theta_x - s \alpha \delta \theta_y)] \\ a [c \phi s \lambda - s \phi s \lambda (s \alpha \delta \theta_x + c \alpha \delta \theta_y) + c \lambda (c \alpha \delta \theta_x - s \alpha \delta \theta_y)] \\ b [s \phi + c \phi (s \alpha \delta \theta_x + c \alpha \delta \theta_y)] \end{bmatrix} \right\} \\ &= C_z^\alpha \left\{ \begin{bmatrix} a_s c \phi_s c \lambda_s \\ a_s c \phi_s s \lambda_s \\ b_s s \phi_s \end{bmatrix} - \begin{bmatrix} a c \phi c \lambda \\ a c \phi s \lambda \\ b s \phi \end{bmatrix} \right\} \\ &\quad - C_z^\alpha \left[ \begin{bmatrix} a [(-s \phi c \lambda s \alpha - s \lambda c \alpha) \delta \theta_x + (-s \phi c \lambda c \alpha + s \lambda s \alpha) \delta \theta_y] \\ a [(-s \phi s \lambda s \alpha + c \lambda c \alpha) \delta \theta_x + (-s \phi s \lambda c \alpha - c \lambda s \alpha) \delta \theta_y] \\ b [(c \phi s \alpha) \delta \theta_x + (c \phi c \alpha) \delta \theta_y] \end{bmatrix} \right] \\ &= \Delta\mathbf{r}^e + \begin{bmatrix} b C_{nxy}^e & -b C_{nxz}^e & 0 \\ a C_{nyy}^e & -a C_{nyx}^e & 0 \\ a C_{nzy}^e & -a C_{nzx}^e & 0 \end{bmatrix} \begin{bmatrix} \delta \theta_x \\ \delta \theta_y \\ \delta \theta_z \end{bmatrix}\end{aligned}$$

where

$$\begin{aligned} a_s &\equiv R_n + h_s \\ b_s &\equiv R_n(1 - \varepsilon^2) + h_s \\ a &\equiv R_n + h \\ b &\equiv R_n(1 - \varepsilon^2) + h \end{aligned}$$

and  $\Delta\mathbf{r}^e$  is the relative position vector without error.

For this illustration, the second partial derivative in Eq. (7.13) is obtained directly from the preceding relative range difference equation:

$$\frac{\partial \Delta\mathbf{r}^e}{\partial \delta\boldsymbol{\theta}} = \begin{bmatrix} bC_{nxy}^e & -bC_{nxz}^e & 0 \\ aC_{nyy}^e & -aC_{nyx}^e & 0 \\ aC_{nzy}^e & -aC_{nzx}^e & 0 \end{bmatrix}$$

The first few terms of the linearized measurement matrix corresponding to the navigation system angular position errors become

$$H_{\Delta\rho} = \frac{\partial \Delta\rho}{\partial \delta\boldsymbol{\theta}, \dots} = \left[ \frac{\partial \Delta\rho}{\partial \delta\theta_x} \quad \frac{\partial \Delta\rho}{\partial \delta\theta_y} \quad \dots \right]$$

## 7.3 Global-Positioning-System Range

### 7.3.1 Global-Positioning-System Functionality

GPS data are provided in a variety of forms. In their most basic form, these data are a transit time difference and Doppler effect—pseudorange and delta range—between a receiver's antenna and GPS satellites. In some GPS receiver units, these basic data are preprocessed to provide refined position and velocity data in either local geodetic coordinates (latitude and longitude) or ECEF coordinates.

The range provided by GPS is functionally similar to that provided by TACAN, except that the TACAN station is fixed to the Earth's surface and GPS satellite positions must be determined in orbit. Satellite positions are assumed to be known relatively accurately. The GPS system assumes the satellite orbital motion is described by Kepler's equations. Corrections are applied to the parameters describing this motion to provide the user with accurate satellite positioning information. A description of this motion is presented, based on Newton's laws of motion.

Consider the central force attraction illustrated in Fig. 7.5. The gravitational attraction is between the two bodies,  $M$  and  $m$ , whose common mass center is centered in the larger of the two masses. Accelerations act on the smaller mass in radial and tangential directions along the orbit's arc. Expressions for this acceleration are developed using polar coordinates.

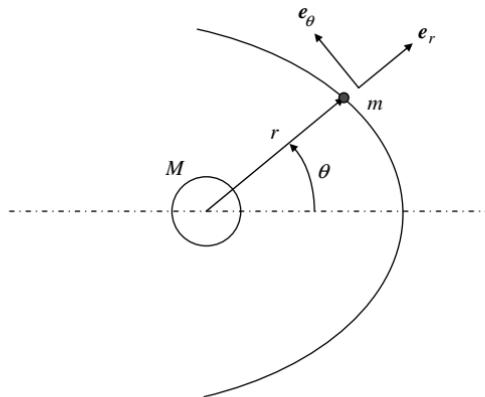


Fig. 7.5 Central force attraction.

Small displacements along the orbit line's arc are shown in Fig. 7.6. From this figure, the rate of change of unit vectors  $\mathbf{e}_r$  and  $\mathbf{e}_\theta$ , along and normal to the position vector  $\mathbf{r}$ , are formed. The differential change in the radial  $\mathbf{e}_r$  and its normal  $\mathbf{e}_\theta$  unit vectors resulting from a differential angle rotation  $d\theta$  from position (1) to position (2) is

$$d\mathbf{e}_r = \mathbf{e}_r^{(2)} - \mathbf{e}_r^{(1)} = d\theta \mathbf{e}_\theta$$

$$d\mathbf{e}_\theta = \mathbf{e}_\theta^{(2)} - \mathbf{e}_\theta^{(1)} = -d\theta \mathbf{e}_r$$

Dividing by differential time  $dt$  yields the time rate of change of these unit vectors:

$$\dot{\mathbf{e}}_r = \dot{\theta} \mathbf{e}_\theta \quad (7.14)$$

$$\dot{\mathbf{e}}_\theta = -\dot{\theta} \mathbf{e}_r \quad (7.15)$$

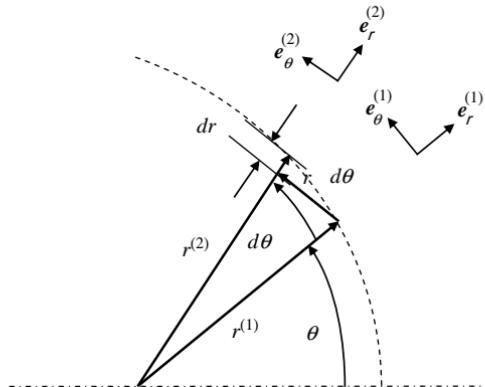


Fig. 7.6 Position change along an arc.

Returning to Fig. 7.5, the position vector is expressed in polar coordinates as

$$\mathbf{r} = r\mathbf{e}_r \quad (7.16)$$

The velocity is obtained from the time derivative of Eq. (7.16) using Eqs. (7.14) and (7.15):

$$\begin{aligned}\mathbf{v} &= \dot{\mathbf{r}} = \dot{r}\mathbf{e}_r + r\dot{\theta}\mathbf{e}_\theta \\ &= \dot{r}\mathbf{e}_r + r\dot{\theta}\mathbf{e}_\theta\end{aligned} \quad (7.17)$$

Taking a second derivative, the acceleration is obtained:

$$\begin{aligned}\mathbf{a} &= \ddot{\mathbf{v}} = \ddot{r}\mathbf{e}_r + \dot{r}\dot{\theta}\mathbf{e}_\theta + \dot{r}\dot{\theta}\mathbf{e}_\theta + r\ddot{\theta}\mathbf{e}_\theta + r\dot{\theta}^2\mathbf{e}_\theta \\ &= \ddot{r}\mathbf{e}_r + \dot{r}\dot{\theta}\mathbf{e}_\theta + \dot{r}\dot{\theta}\mathbf{e}_\theta + r\ddot{\theta}\mathbf{e}_\theta - r\dot{\theta}^2\mathbf{e}_r \\ &= (\ddot{r} - r\dot{\theta}^2)\mathbf{e}_r + (2\dot{r}\dot{\theta} + r\ddot{\theta})\mathbf{e}_\theta\end{aligned} \quad (7.18)$$

It is assumed that the acceleration  $\mathbf{a}$  is the result of only central force attraction acting along the radius vector, as shown in Fig. 7.5, from the two-body system's common center of mass. This force is expressed as

$$\mathbf{F} = -\frac{\mu m}{r^2}\mathbf{e}_r \quad (7.19)$$

where  $\mu$  is the gravitational constant of the attracting mass  $M$  and  $m$  the satellite mass.

The force per unit mass, specific force (acceleration), from Eq. (7.19) is set equal to Eq. (7.18), and common unit vector components are grouped resulting in the following equations of motion:

$$\ddot{r} - r\dot{\theta}^2 = -\frac{\mu}{r^2} \quad (7.20)$$

and

$$r\ddot{\theta} + 2\dot{r}\dot{\theta} = 0 \quad (7.21)$$

The solution to Eq. (7.20) can be obtained with the change of variable  $-1/u = r$ . This solution, position in an elliptical orbit and the corresponding velocity, is presented in Appendix B. With reference to definitions of the satellite orbit presented in Figs. 7.7 and 7.8, Table 7.1 summarizes the solution to Eqs. (7.20) and (7.21) for an elliptical orbit.

The GPS satellite position is defined in terms of the orbital elements defined in Figs. 7.7 and 7.8. In the primed axis system defined in the orbital plane,

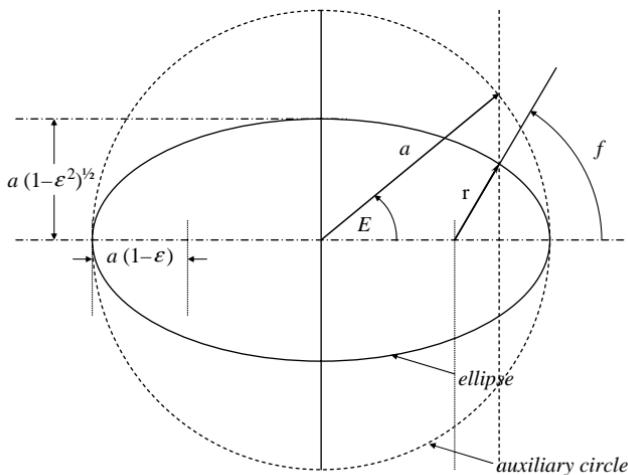


Fig. 7.7 Orbital plane elements.

the  $x'$  and  $y'$  positions are

$$x' = r \cos \theta \quad (7.22)$$

$$y' = r \sin \theta \quad (7.23)$$

These positions are transformed into an Earth-centered frame, but, first, it is easier to form the transformation from Earth-centered to the orbital plane, and then transpose this result to obtain the transformation from orbital plane to Earth-centered. Referring to Fig. B.1, the rotation sequence follows first with

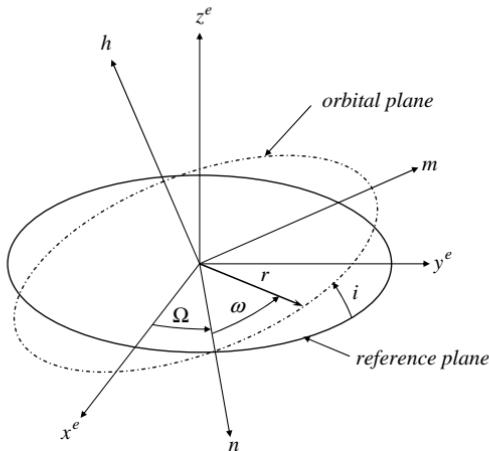


Fig. 7.8 Satellite ECEF positions.

Table 7.1 Satellite orbit equations

Orbit element	Equation
Mean motion	$n = 2\pi/\tau = \sqrt{\mu/a^3}$
Mean anomaly	$M = (2\pi/\tau)t$
Eccentric anomaly $E$	$E = \varepsilon \sin E + M$
True anomaly $f$	$\cos f = \cos E - \varepsilon/1 - \varepsilon \cos E$
Orbit radius $r$	$r = a(1 - \varepsilon \cos E)$
Latitude $\theta$	$\theta = \omega + f$
Longitude $\Omega$	$\Omega = \Omega_0 + (\dot{\Omega} - \dot{\Omega}_{i/e})t$
Orbital elements (see Figs. 7.7 and 7.8)	—
Semimajor axis $a$	
Eccentricity $\varepsilon$	
Inclination with respect to equator $I$	
Longitude of ascending node $\Omega$	
Argument of perigee from the reference $\omega$	

the longitude, then the inclination as

$$\begin{bmatrix} x' \\ y' \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & ci & si \\ 0 & -si & ci \end{bmatrix} \begin{bmatrix} c\Omega & s\Omega & 0 \\ -s\Omega & c\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x^e \\ y^e \\ z^e \end{bmatrix} \quad (7.24)$$

which, when the product of the matrices are transposed, yields the transformation from orbital plane to Earth-centered as

$$\begin{aligned} \begin{bmatrix} x^e \\ y^e \\ z^e \end{bmatrix} &= \begin{bmatrix} c\Omega & -s\Omega ci & s\Omega si \\ s\Omega & c\Omega ci & -c\Omega si \\ 0 & si & ci \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} c\Omega x' - cis\Omega y' \\ s\Omega x' + cic\Omega y' \\ si y' \end{bmatrix} \end{aligned} \quad (7.25)$$

Or, combining Eqs. (7.22), (7.23), and (7.25),

$$x^{\text{ECEF}} = r[\cos \theta \cos \Omega - \sin \theta \cos i \sin \Omega] \quad (7.26)$$

$$y^{\text{ECEF}} = r[\cos \theta \sin \Omega + \sin \theta \cos i \cos \Omega] \quad (7.27)$$

$$z^{\text{ECEF}} = r \sin \theta \sin i \quad (7.28)$$

Orbital parameters, including the longitude of the ascending node  $\Omega$ , vary as a result of the Earth's gravitational variation. The mean rate of change of this parameter, assuming the gravitational terms including the first nonzero term in the

potential function, is found as

$$\frac{d\bar{\Omega}}{dt} = -\frac{3}{2} J_2 \left( \frac{r_e}{p} \right)^2 n \cos i \quad (7.29)$$

Because this is an idealization, the actual gravitational variation will cause change to the orbital parameters in addition to that predicted by this expression, thus requiring updates to the simplified equations in Table 7.1. The longitude of the ascending node is predicted by the following expression

$$\Omega = \Omega_0 + (\dot{\Omega} - \dot{\Omega}_{i/e})t \quad (7.30)$$

Corrections to this simple description of satellite motion are provided by GPS ground-based satellite monitoring. Updates to the satellite orbital elements, in the form of corrections, are provided for mean motion  $n$ , argument of latitude  $\theta$ , radius  $r$ , inclination  $i$ , and longitude of ascending node  $\Omega$ .

### 7.3.2 Satellite Selection

Normally, several satellites are within view of the GPS receiver's antenna. However, better performance can be obtained by properly selecting which of the satellites is to be used. Satellites are selected based on their elevation above the local horizon and their geometrical relationship to the user.

**Elevation angle.** Of the satellites available, some are not visible or sufficiently high above the horizon to be used. Range data from satellites that are near the horizon are corrupted by atmospheric refraction and should be excluded from use. To select which satellites will be used, the elevation angle, shown in Fig. 7.9, from the receiver's antenna position to each candidate satellite is determined.

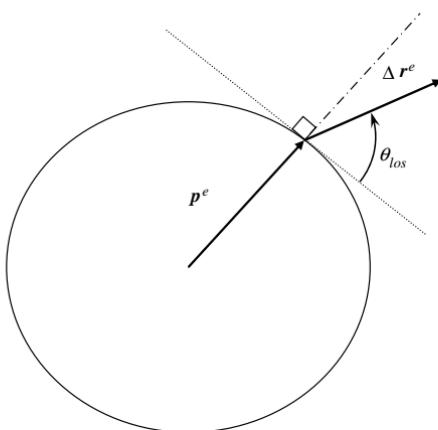


Fig. 7.9 Satellite LOS elevation angle.

The LOS unit vector in ECEF from the receiver's antenna position to the satellite position is

$$\mathbf{e}_{\text{los}}^e = \frac{\Delta \mathbf{r}^e}{\rho} \quad (7.31)$$

The local vertical unit vector can be approximated as

$$\mathbf{e}_{\text{up}}^e \approx \frac{\mathbf{p}_a^e}{R} \quad (7.32)$$

where the magnitude of the receiver's antenna ECEF position vector is

$$|\mathbf{p}_a^e| \equiv R = \sqrt{p_1^2 + p_2^2 + p_3^2} \quad (7.33)$$

Using the dot-product relationship for two unit vectors, the elevation angle relative to the local horizontal is determined from

$$\sin \theta_{\text{los}} = \mathbf{e}_{\text{los}}^e \cdot \mathbf{e}_{\text{up}}^e \quad (7.34)$$

Satellites that are too near the horizon are usually excluded from use in the measurement processing because of the atmosphere distorting the range information.

*Geometric dilution of precision.* In addition to selecting a satellite based in its elevation angle, satellites can be selected based on their geometrical relationship to the GPS receiver's antenna, as shown in Fig. 7.10. The geometry of the satellite positions relative to the antenna determines the GDOP or loss of precision resulting from this geometry. The minimum GDOP value is considered

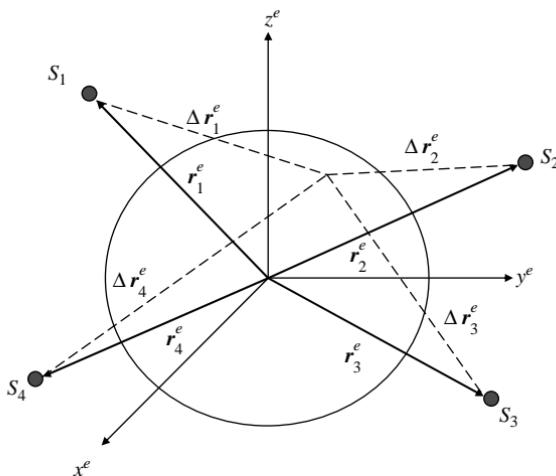


Fig. 7.10 Multiple-satellite geometry.

the best. For a selected group of satellites, the minimum GDOP is used to determine which group of satellites to use in the measurement processing.

The  $i$ th satellite's LOS unit vector  $\mathbf{e}_{\text{los}_i}^L$ , with respect to the local horizontal navigation frame, is obtained from

$$\mathbf{e}_{\text{los}}^L = C_e^L \mathbf{e}_{\text{los}}^e \quad (7.35)$$

where the sub- and superscripts have been momentarily dropped.

Consider the case where four satellites are within view:  $i = 1, 2, 3, 4$ . A Kalman filter's use of satellite-range and/or delta-range updates can accommodate as few as one satellite, and, of course, more than four. The pseudoranges from the receiver's antenna to the four satellites are computed from the component differences of the position vector  $\mathbf{p}$  and each of the satellite's position vectors  $\mathbf{s}$ . Assuming all ranges are biased by the same receiver clock error, the  $i$ th pseudorange is expressed (neglecting additive noise) as

$$\rho_i = [(x_i - p_1)^2 + (y_i - p_2)^2 + (z_i - p_3)^2]^{1/2} + b \quad (7.36)$$

where  $x$ ,  $y$ , and  $z$  represent the vector components of the  $i$ th satellite's position vector  $\mathbf{s}$ . If the receiver's antenna position is in error by  $\delta\mathbf{p}$ , then the linearized form of this equation becomes

$$\delta\rho_i \approx -\frac{[(x_i - p_1)\delta p_1 + (y_i - p_2)\delta p_2 + (z_i - p_3)\delta p_3]}{\rho_i} + \delta b \quad (7.37)$$

The differences in vector components divided by the magnitude is the unit vector

$$\mathbf{e}_i = \begin{bmatrix} x_i - p_1 \\ y_i - p_2 \\ z_i - p_3 \end{bmatrix} \frac{1}{\rho_i} \quad (7.38)$$

For four satellites, the variation in pseudorange can be expressed as

$$\begin{bmatrix} \delta\rho_1 \\ \delta\rho_2 \\ \delta\rho_3 \\ \delta\rho_4 \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1^T & 1 \\ \mathbf{e}_2^T & 1 \\ \mathbf{e}_3^T & 1 \\ \mathbf{e}_4^T & 1 \end{bmatrix} \quad (7.39)$$

This linear vector equation can be redefined in the following form:

$$\mathbf{z} = H\mathbf{x} \quad (7.40)$$

where the vector  $\mathbf{x}$  contains the corrections to position and time.

If the matrix  $H$  is nonsingular, then this equation can be solved for  $\mathbf{x}$  as

$$\mathbf{x} = H^{-1}\mathbf{z} \quad (7.41)$$

Assuming that unknown corrections to position and time are random, resulting from variable satellite positions, the expected error in position and time is given as its covariance

$$\begin{aligned}\text{Cov}[\mathbf{x}] &\equiv E[\mathbf{x}\mathbf{x}^T] \\ &= E[(H^{-1}\mathbf{z})(H^{-1}\mathbf{z})^T] \\ &= H^{-1}E[\mathbf{z}\mathbf{z}^T]H^{-T}\end{aligned}\quad (7.42)$$

If the pseudorange error variance is the same for the four satellites and is equal to  $\sigma_{\text{pseudo}}^2$ , then

$$E[\mathbf{z}\mathbf{z}^T] = \sigma_{\text{pseudo}}^2 I_{4 \times 4} \quad (7.43)$$

Using this in Eq. (7.42), the covariance of position and time corrections becomes

$$\text{Cov}[\mathbf{x}] = (H^T H)^{-1} \sigma_{\text{pseudo}}^2 \quad (7.44)$$

where it is assumed that pseudorange error statistics are the same. The error associated with the position and time corrections is governed by the geometry contained in the matrix  $H$ .

Reinstating sub- and superscripts again and defining the matrix  $(\gamma)$ , Eq. (7.44) is rewritten as

$$(\gamma) = \begin{bmatrix} \mathbf{e}_{\text{los}1}^{L^T} & 1 \\ \mathbf{e}_{\text{los}2}^{L^T} & 1 \\ \mathbf{e}_{\text{los}3}^{L^T} & 1 \\ \mathbf{e}_{\text{los}4}^{L^T} & 1 \end{bmatrix} \quad (7.45)$$

From the matrix

$$\Xi = [(\gamma)^T (\gamma)]^{-1} \quad (7.46)$$

Geometric dilution of precision is computed from its diagonals as

$$\text{GDOP} = \sqrt{\Xi_{1,1} + \Xi_{2,2} + \Xi_{3,3} + \Xi_{4,4}} \quad (7.47)$$

position dilution of precision (PDOP) as

$$\text{PDOP} = \sqrt{\Xi_{1,1} + \Xi_{2,2} + \Xi_{3,3}} \quad (7.48)$$

and finally, horizontal dilution of precision (HDOP) as

$$\text{HDOP} = \sqrt{\Xi_{1,1} + \Xi_{2,2}} \quad (7.49)$$

GPS pseudorange and delta range measurements and their linearizations suitable for filter implementation are summarized in Fig. 7.11.

Pseudorange:

$$\rho = |\Delta r| + b$$

$$H_{\text{pseudo}} = \begin{bmatrix} e^e_{\text{LOS}}^T & | & O_{I \times 3} & | & O_{I \times 3} & | & 0 & 1 & 0 \end{bmatrix}$$

Delta Range/Pseudo range Rate:

$$\Delta\rho = r_f - r_b + d \Delta t$$

$$H_{\text{delt}} = \begin{bmatrix} O_{I \times 3} & | & e^e_{\text{LOS}}^T \Delta t & | & O_{I \times 3} & | & 0 & 0 & \Delta t \end{bmatrix}$$

where

$r$  - range from vehicle to satellite

$e^e_{\text{LOS}}$  - unit line of sight vector from vehicle to satellite

$\Delta t$  - difference between final time  $t_f$  and beginning time  $t_b$  for Doppler integration interval

Fig. 7.11 GPS receiver measurement model.

## 7.4 Forward-Looking Infrared Line-of-Sight Systems

### 7.4.1 Line-of-Sight System Functionality

Line-of-sight systems include airborne radar, FLIR, TV, laser ranging, etc. These systems establish LOS to a feature relative to the vehicle's axis system (assumed to be parallel with the system body axis), as illustrated in Fig. 7.12.

Generally, these systems employ two or more mechanical gimbals, within which the sensor moves to establish the LOS. The outer gimbal (first movement relative to the aircraft body axis) is usually azimuth  $Az$ , and the inner gimbal (mounted within the outer gimbal) is usually elevation  $El$ . The gimbal ordering, and therefore the sequence of angle rotations, is important to establishing the transformation matrix that transforms vector components from the LOS to the aircraft axis system and visa versa. This then forms the basis of the mathematical model for these measurements.

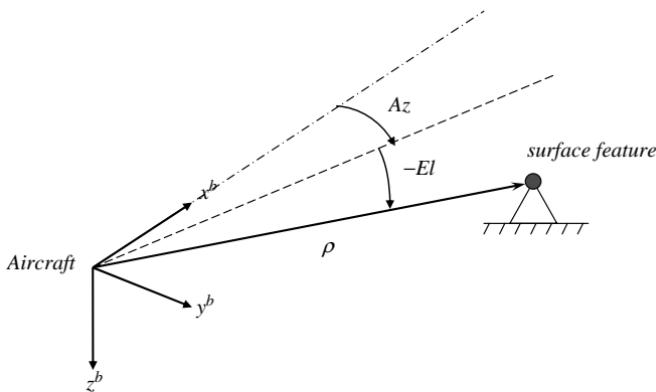


Fig. 7.12 LOS to surface feature.

### 7.4.2 LOS System Measurements

If LOS is established by the sensor, all components of the relative position vector are zero except for the first, which is the magnitude of the relative range between the aircraft and the ground surface feature

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}^{los} \equiv \begin{bmatrix} \rho \\ 0 \\ 0 \end{bmatrix} \quad (7.50)$$

In the following, it is assumed that the gimbal arrangement for LOS systems is such that the gimbal mounted to the aircraft is azimuth. The next inner gimbal is elevation. Transforming from the body axis to the LOS axis proceeds as follows:

$$\begin{aligned} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}^{los} &= \begin{bmatrix} cEl & 0 & -sEl \\ 0 & 1 & 0 \\ sEl & 0 & cEl \end{bmatrix} \begin{bmatrix} cAz & sAz & 0 \\ -sAz & cAz & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}^b \\ &= \begin{bmatrix} cElcAz & cElsAz & -sEl \\ -sAz & cAz & 0 \\ sElcAz & sElsAz & cEl \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}^b \end{aligned} \quad (7.51)$$

Equations for azimuth and elevation in terms of the body-referenced relative position vector components are obtained by equating the right-hand sides of Eqs. (7.50) and (7.51). The resulting measurement equations and their corresponding linearizations for the matrix, that is,

$$\frac{\partial \Delta Az}{\partial \Delta r^b} \quad (7.52)$$

and

$$\frac{\partial \Delta El}{\partial \Delta r^b} \quad (7.53)$$

are presented in Table 7.2 for this LOS system.

### Example 7.3 Complete Line-of-Sight Measurement Formulation

The use of LOS measurements couples the navigation system position error, as examined earlier for TACAN measurements and navigation system attitude errors, into the measurement process. As with TACAN, the relative position vector between the navigation system's position, and the ground reference point is expressed as

$$\Delta \bar{r}^b = \bar{C}_n^b \bar{C}_e^n C_z^x \left\{ \begin{bmatrix} (R_n + h_s) \cos \phi_s \cos \lambda_s \\ (R_n + h_s) \cos \phi_s \sin \lambda_s \\ [R_n(1 - \varepsilon^2) + h_s] \sin \phi_s \end{bmatrix} - \begin{bmatrix} (R_n + h) \cos \bar{\phi} \cos \bar{\lambda} \\ (R_n + h) \cos \bar{\phi} \sin \bar{\lambda} \\ [R_n(1 - \varepsilon^2) + h] \sin \bar{\phi} \end{bmatrix} \right\}$$

Table 7.2 LOS system measurements and linearizations

Measurement	Partial derivative with respect to ( )		
	$\Delta x$	$\Delta y$	$\Delta z$
$Az = \tan^{-1}\left(\frac{\Delta y}{\Delta x}\right)$	$\frac{-\Delta y}{\Delta x^2 + \Delta y^2}$	$\frac{\Delta x}{\Delta x^2 + \Delta y^2}$	
$El = \tan^{-1}\left(\frac{-\Delta z}{\sqrt{\Delta x^2 + \Delta y^2}}\right)$	$\frac{\Delta x \Delta z}{\rho^2 \sqrt{\Delta x^2 + \Delta y^2}}$	$\frac{\Delta y \Delta z}{\rho^2 \sqrt{\Delta x^2 + \Delta y^2}}$	$\frac{-(\Delta x^2 + \Delta y^2)}{\rho^2 \sqrt{\Delta x^2 + \Delta y^2}}$
$\rho = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$	$\frac{\Delta x}{\rho}$	$\frac{\Delta y}{\rho}$	$\frac{\Delta z}{\rho}$

where

$$\begin{aligned}\bar{C}_n^b \bar{C}_e^n &= (\bar{C}_n^e \bar{C}_b^n)^T \\ &= (C_n^e [I + (\delta\theta \times)] [I - (\phi x)] C_b^n)^T \\ &\approx C_n^b [I + (\phi - \delta\theta) \times] C_e^n\end{aligned}$$

The navigation system computed latitude and longitude terms in the brackets in the preceding equation were expanded in Example 7.2. Using these results, the following is obtained for the body-frame relative position vector:

$$\Delta\Delta\mathbf{r}^b = \Delta\mathbf{r}^b + C_n^b[(\phi - \delta\theta) \times] C_e^n \Delta\mathbf{r}^e + C_n^b C_e^n \begin{bmatrix} bC_{nxy}^e & -bC_{nxz}^e & 0 \\ aC_{nyy}^e & -aC_{nyx}^e & 0 \\ aC_{nzy}^e & -aC_{nzx}^e & 0 \end{bmatrix} \delta\theta$$

Recognizing that

$$\Delta\mathbf{r}^n = C_e^n \Delta\mathbf{r}^e$$

and exchanging the order of the vector cross-product terms in the second term in this equation, the following is obtained for the relative position vector:

$$\Delta\Delta\mathbf{r}^b = \Delta\mathbf{r}^b - C_n^b(\Delta\mathbf{r}^n \times) \phi + C_n^b C_e^n \left\{ \begin{bmatrix} bC_{nxy}^e & -bC_{nxz}^e & 0 \\ aC_{nyy}^e & -aC_{nyx}^e & 0 \\ aC_{nzy}^e & -aC_{nzx}^e & 0 \end{bmatrix} + (\Delta\mathbf{r}^n \times) \right\} \delta\theta$$

The partial derivatives to accompany those in Eqs. (7.52) and (7.53), with respect to the angular position error and the attitude error, are

$$\frac{\partial \Delta\mathbf{r}^b}{\partial \delta\theta} = C_n^b C_e^n \left\{ \begin{bmatrix} bC_{nxy}^e & -bC_{nxz}^e & 0 \\ aC_{nyy}^e & -aC_{nyx}^e & 0 \\ aC_{nzy}^e & -aC_{nzx}^e & 0 \end{bmatrix} + (\Delta\mathbf{r}^n \times) \right\}$$

and

$$\frac{\partial \Delta \mathbf{r}^b}{\partial \boldsymbol{\phi}} = -C_n^b (\Delta \mathbf{r}^n \times)$$

To complete the measurement linearization, partial derivatives of azimuth (and similarly for elevation) LOS measurement, with respect to angular position and attitude errors, are obtained from the following:

$$\begin{aligned}\frac{\partial \Delta A_z}{\partial \boldsymbol{\delta \theta}} &= \frac{\partial \Delta A_z}{\partial \Delta \mathbf{r}^b} \frac{\partial \Delta \mathbf{r}^b}{\partial \boldsymbol{\delta \theta}} \\ \frac{\partial \Delta A_z}{\partial \boldsymbol{\phi}} &= \frac{\partial \Delta A_z}{\partial \Delta \mathbf{r}^b} \frac{\partial \Delta \mathbf{r}^b}{\partial \boldsymbol{\phi}}\end{aligned}$$

From this example, it can be seen that errors in LOS navigation aids include errors in body attitudes, in addition to position errors.

## 7.5 Chapter Summary

In this chapter, several navigation aids providing independent and/or redundant navigation state data, operation, and performance characteristics were presented. Their error characterizations and examples of combining outputs as measurements into an integrated navigation system were presented using several navigation state error representations from Chapter 5. Examples presented were DVS, TACAN, GPS, and FLIR.

Additionally, TACAN was used to illustrate a single ranging operation and associated error modeling. This error model is applied later in Chapter 10 for postflight trajectory reconstruction. TACAN also functions as an example of how single-satellite ranging can be used in GPS applications. Single-satellite GPS operations are used in a case study presented in Chapter 12 for a GPS/inertial integrated navigation system. Extensions for multiple-satellite ranging for GPS were presented, along with GPS quality measures of satellite elevation and GDOP, which are also illustrated in this later case study.

Mathematical methods presented in this chapter can be applied to other navigation aids and other navigation state error representations. It is the methodology, not specific examples presented, that is important because other navigation aids or different navigation state error representations can confront the integrator/designer applications.

## Problems

- 7.1** Doppler divergence (alternative velocity error representation) is dealt with in this exercise. Express Eq. (7.10) in the navigation frame  $n$  and expand about a nominal velocity  $\mathbf{v}^n$ , but this time use the alternate representation for INU velocity error as

$$\begin{aligned}\Delta \mathbf{v}_{\text{DOP}}^n &= \tilde{\mathbf{v}}_{\text{INU}}^n - \bar{\mathbf{v}}_{\text{DOP}}^n \\ &= \mathbf{v}_{\text{INU}}^n + \delta \mathbf{v}_{\text{INU}}^1 - (\boldsymbol{\delta \theta} \times) \mathbf{v}_{\text{INU}}^n - \tilde{C}_b^n (\mathbf{v}_{\text{DOP}}^b + \delta \mathbf{v}_{\text{DOP}}^b)\end{aligned}$$

$$\begin{aligned}
&= \mathbf{v}_{\text{INU}}^n + \delta\mathbf{v}_{\text{INU}}^1 - (\delta\boldsymbol{\theta} \times) \mathbf{v}_{\text{INU}}^n - [I - (\boldsymbol{\phi} \times)] C_b^n (\mathbf{v}_{\text{DOP}}^b + \delta\mathbf{v}_{\text{DOP}}^b) \\
&\approx \mathbf{v}_{\text{INU}}^n - C_b^n \mathbf{v}_{\text{DOP}}^b + \delta\mathbf{v}_{\text{INU}}^1 - (\delta\boldsymbol{\theta} \times) \mathbf{v}_{\text{INU}}^n - C_b^n \delta\mathbf{v}_{\text{DOP}}^b + (\boldsymbol{\phi} \times) C_b^n \mathbf{v}_{\text{DOP}}^b \\
&= \delta\mathbf{v}^1 + [(\boldsymbol{\phi} - \delta\boldsymbol{\theta}) \times] C_b^n \mathbf{v}^b - C_b^n \delta\mathbf{v}_{\text{DOP}}^b \\
&= \delta\mathbf{v}^1 - (\mathbf{v}_{\text{INU}}^n \times) \boldsymbol{\psi} - C_b^n \delta\mathbf{v}_{\text{DOP}}^b
\end{aligned}$$

Form the first few terms of the linearized measurement matrix corresponding to the navigation system errors and Doppler errors for the computer frame referenced velocity error as

$$\begin{aligned}
H_{\text{DOP}} &= \frac{\partial \Delta\mathbf{v}}{\partial \delta\boldsymbol{\theta}, \delta\mathbf{v}^1, \boldsymbol{\psi}, \dots, \delta\mathbf{v}_{\text{DOP}}^b} \\
&= \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} & -(\mathbf{v}^n \times) & \cdots & -C_b^n \end{bmatrix}
\end{aligned}$$

- 7.2 Consider the alternate velocity doppler/INU measurement  $\boldsymbol{\phi}$  formulation. Specialize the INU/Doppler difference in Sec. 7.1 for the alternative velocity error representation to the following. Assume the position error is linear position and expressed as

$$\delta\boldsymbol{\theta} \equiv \begin{bmatrix} -\frac{\delta r_y}{R} \\ \frac{\delta r_x}{R} \\ \delta\theta_z \end{bmatrix}$$

The attitude error is the  $\boldsymbol{\phi}$  form. Show that the velocity difference can be expressed as

$$\Delta\mathbf{v}^n = \begin{bmatrix} -\frac{v_z}{R} & 0 & 0 & 1 & 0 & 0 & 0 & v_z & -v_y & v_y \\ 0 & -\frac{v_z}{R} & 0 & 0 & 1 & 0 & -v_z & 0 & v_x & -v_x \\ \rho_y & -\rho_x & 0 & 0 & 0 & 1 & v_y & -v_x & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta h \\ \delta v_x^1 \\ \delta v_y^1 \\ \delta v_z^1 \\ \phi_x \\ \phi_y \\ \phi_z \\ \delta\theta_z \end{bmatrix} - C_b^n \delta\mathbf{v}_{\text{DOP}}^b$$

- 7.3 Consider the alternate velocity doppler/INU measurement  $\psi$  formulation. Again, specialize the INU/Doppler difference in Sec. 7.1 to the following. Assume the position error is the linear position as just shown, the velocity error is the alternate error form, and the attitude error is the  $\psi$  form. Show that the velocity difference can be expressed as

$$\Delta \mathbf{v}^n = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & v_z & -v_y \\ 0 & 0 & 0 & 0 & 1 & 0 & -v_z & 0 & v_x \\ 0 & 0 & 0 & 0 & 0 & 1 & v_y & -v_x & 0 \end{bmatrix} \begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta h \\ \delta v_x^1 \\ \delta v_y^1 \\ \delta v_z^1 \\ \psi_x \\ \psi_y \\ \psi_z \end{bmatrix} - C_b^n \delta \mathbf{v}_{\text{DOP}}^b$$

- 7.4 For the preceding two specializations of the INU/Doppler velocity difference, the Doppler errors were not specialized. Assume that the vehicle is a fixed-wing aircraft whose velocity is primarily the  $x$ -axis body component and that the other components, when multiplied by Doppler errors, are assumed negligible. Show that the Doppler contribution to these two specializations is given as

$$C_b^n \delta \mathbf{v}_{\text{DOP}}^b = C_b^n \begin{bmatrix} v_x^b & 0 & 0 & 1 & 0 & 0 \\ 0 & v_x^b & 0 & 0 & 1 & 0 \\ 0 & 0 & v_x^b & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dSF_x \\ dMA_{yx} \\ dMA_{zx} \\ dB_x \\ dB_y \\ dB_z \end{bmatrix}$$

- 7.5 The Doppler velocity difference in Eq. (7.10) can also be formed in body axes. Reformulate the Doppler divergence in a body-frame using the perturbation form of velocity error representation.

$$\begin{aligned} \Delta \mathbf{v}_{\text{DOP}}^b &= \bar{\mathbf{v}}_{\text{INU}}^b - \bar{\mathbf{v}}_{\text{DOP}}^b \\ &= \bar{C}_n^b \bar{\mathbf{v}}_{\text{INU}}^n - (\mathbf{v}_{\text{DOP}}^b + \delta \mathbf{v}_{\text{DOP}}^b) \\ &= [I + (\boldsymbol{\phi} \times)] C_n^b (\mathbf{v}_{\text{INU}}^n + \delta \mathbf{v}_{\text{INU}}^n) - (\mathbf{v}_{\text{DOP}}^b + \delta \mathbf{v}_{\text{DOP}}^b) \\ &\approx C_n^b \mathbf{v}_{\text{INU}}^n - \mathbf{v}_{\text{DOP}}^b + C_n^b \delta \mathbf{v}_{\text{INU}}^n + (\boldsymbol{\phi} \times) C_n^b \mathbf{v}_{\text{INU}}^n - \delta \mathbf{v}_{\text{DOP}}^b \\ &= C_n^b \delta \mathbf{v}_{\text{INU}}^n + (\boldsymbol{\phi} \times) C_n^b \mathbf{v}_{\text{INU}}^n - \delta \mathbf{v}_{\text{DOP}}^b \\ &= C_n^b \delta \mathbf{v}_{\text{INU}}^n - (\mathbf{v}_{\text{INU}}^b \times) \boldsymbol{\phi} - C_b^n \delta \mathbf{v}_{\text{DOP}}^b \end{aligned}$$

- 7.6 Consider GPS/INU angular position differences from latitude/longitude differences, and reexamine GPS position differences in Problem 5.8. Assume that latitude and longitude differences  $\Delta\phi$  and  $\Delta\lambda$

are obtained from

$$\begin{aligned}\Delta\phi &= \phi_{\text{INU}} - \phi_{\text{GPS}} \\ \Delta\lambda &= \lambda_{\text{INU}} - \lambda_{\text{GPS}}\end{aligned}$$

and the INU wander angle  $\alpha$  is used to transform these differences into the navigation frame. Show the following results:

$$\Delta\boldsymbol{\theta}^n = \delta\boldsymbol{\theta}^n + \begin{bmatrix} \delta\alpha\delta\theta_y \\ -\delta\alpha\delta\theta_x \\ 0 \end{bmatrix}$$

- 7.7** Review GPS/INU navigation frame velocity differences: alternate velocity. Reexamine GPS velocity differences. Assume GPS velocity available in the ECEF frame without error. For perturbation and alternative velocity error representations, show that the difference

$$\Delta\mathbf{v} = \bar{\mathbf{v}}_{\text{INU}}^n - \bar{C}_e^n \mathbf{v}_{\text{GPS}}^e$$

yields for the following:

Perturbation:

$$\Delta\mathbf{v} = \delta\mathbf{v}^n - (\mathbf{v}^n \times) \delta\boldsymbol{\theta}$$

Alternative:

$$\Delta\mathbf{v} = \delta\mathbf{v}^1$$

- 7.8** Consider the difference between velocity referenced at a GPS antenna and the navigation system velocity during maneuvers. Specialize the GPS velocity at the antenna by again considering a fixed-wing aircraft. Also assume that the aircraft's body-to-navigation frame rotation is much greater than its transport rate, yielding the following approximation:

$$\begin{aligned}\Delta\mathbf{v}_{\text{INU}/\text{ant}}^n &= \mathbf{v}_{\text{INU}}^n - \mathbf{v}_{\text{ant}}^n \\ &= \boldsymbol{\omega}_{n/b}^b \times (C_b^n \mathbf{d}^b)\end{aligned}$$

For a planar horizontal turn, such that all but the  $z$  component of the preceding rotation vector elements are zero, the velocity difference becomes

$$\begin{aligned}\Delta\mathbf{v}_{\text{INU}/\text{ant}}^n &= \begin{bmatrix} 0 & -\omega_z & 0 \\ \omega_z & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} C_{xx} & - & - \\ C_{yx} & - & - \\ C_{zx} & - & - \end{bmatrix} \begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} -\omega_z C_{yx} d \\ \omega_z C_{xx} d \\ 0 \end{bmatrix}\end{aligned}$$

With the aircraft maintaining constant speed,  $V \approx v_x^b$  and constant bank angle  $\phi$ , the turn rate is

$$\dot{\sigma} = \frac{g}{V} \tan \phi$$

which is approximately equal to  $\omega_z$ . Then the magnitude of the velocity difference is approximately

$$|\Delta \mathbf{v}_{\text{INU/ant}}| = \dot{\sigma} d$$

- 7.9** Specialize the line-of-sight position difference assuming a linear position error and a  $\psi$  attitude error. Express the results in the form of the following linearized measurement matrix:

$$H_{\Delta r} = \frac{\partial \Delta \mathbf{r}}{\partial \delta r(x,y), \delta v, \psi, \delta \theta_z}$$

to obtain

$$H_{\Delta r} = \begin{bmatrix} 1 + \frac{\Delta r_z}{R} & 0 & 0 & 0 & 0 & 0 & \Delta r_z & -\Delta r_y & -\Delta r_y \\ 0 & 1 + \frac{\Delta r_z}{R} & 0 & 0 & 0 & -\Delta r_z & 0 & \Delta r_x & \Delta r_x \\ -\frac{\Delta r_x}{R} & -\frac{\Delta r_y}{R} & 1 & 0 & 0 & 0 & \Delta r_y & -\Delta r_x & 0 & 0 \end{bmatrix}$$

- 7.10** Terrestrial navigation systems also use stellar observations for update. This exercise forms the difference between a body-fixed (not gimbaled) star tracker sensor's outputs and known stellar reference positions to be used as a measurement. Consider the difference vector between onboard computed position, using system variables, and the sensor's outputs (ST):

$$\Delta \mathbf{s}^s = \bar{C}_i^s \mathbf{s}_*^i - \bar{s}_{\text{ST}}^s$$

From star charts or similar information, the inertial position of the star  $\mathbf{s}_*^i$  is assumed known. Expand this difference equation as

$$\Delta \mathbf{s}^s = \bar{C}_b^s \bar{C}_n^b \bar{C}_e^n C_i^e \mathbf{s}_*^i - \bar{s}_{\text{ST}}^s$$

The ST alignment with respect to the body axis is assumed to be represented by

$$\bar{C}_b^s \equiv [I + (\boldsymbol{\delta} \times)] C_b^s$$

where the vector  $\boldsymbol{\delta}$  includes misalignments between the ST and body axes. The other matrices in the vector difference equation were given earlier.

Expand the preceding difference vector to obtain the following:

$$\Delta s^s \approx s_*^s - s_{\text{ST}}^s - (s_*^s \times) [C_n^s \psi + \delta]$$

- 7.11** The mean rate of change of an elliptical orbit's longitude of the ascending node is considered in this exercise [10]. Consider the vector form of the angular momentum (see Appendix B).

$$\begin{aligned} \mathbf{h} = \mathbf{r} \times \mathbf{v} &= \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} -r_z v_y + r_y v_z \\ r_z v_x - r_x v_z \\ -r_y v_x + r_x v_y \end{bmatrix} \\ &= \begin{bmatrix} \frac{r\mu}{h} \sin \Omega \sin i (1 + e \cos f) \\ -\frac{r\mu}{h} \cos \Omega \sin i (1 + e \cos f) \\ \frac{r\mu}{h} \cos i (1 + e \cos f) \end{bmatrix} = \begin{bmatrix} h \sin \Omega \sin i \\ -h \cos \Omega \sin i \\ h \cos i \end{bmatrix} \end{aligned}$$

The rate of change of the angular momentum is obtained as

$$\frac{d}{dt} \mathbf{h} = \frac{d}{dt} (\mathbf{r} \times \mathbf{v}) = \frac{d\mathbf{r}}{dt} \times \mathbf{v} + \mathbf{r} \times \frac{d\mathbf{v}}{dt} = \mathbf{r} \times \frac{d\mathbf{v}}{dt}$$

Because, in an inertial frame, the vectors  $d\mathbf{r}/dt$  and  $\mathbf{v}$  are the same vector and parallel, and the cross-product zero. The specific forces are resolved into component directions that are radial, tangential, and normal to the orbit

$$\frac{d\mathbf{v}}{dt} \equiv \mathbf{f} \equiv R\hat{\mathbf{e}}_R + T\hat{\mathbf{e}}_T + N\hat{\mathbf{e}}_N$$

defining the orbital position vector as

$$\mathbf{r} \equiv r\hat{\mathbf{e}}_R$$

Obtain the following for the terms on the right-hand side of the angular momentum derivative equation:

$$\mathbf{r} \times \frac{d\mathbf{v}}{dt} = -rN\hat{\mathbf{e}}_T + rT\hat{\mathbf{e}}_N$$

Next, obtain the following equation for the derivative of longitude of the ascending node from the first two elements of the preceding angular momentum equation:

$$\dot{\Omega} = \frac{h_x \dot{h}_y - h_y \dot{h}_x}{h_x^2 + h_y^2}$$

Equate the rate of change of angular momentum and specific forces to obtain the following equations:

$$\begin{aligned}\dot{h}_x &= rT(\sin \Omega \sin i) - rN(-\cos \Omega \sin \theta - \sin \Omega \cos \theta \cos i) \\ &= r[(\sin \Omega \sin i)T + (\cos \Omega \sin \theta + \sin \Omega \cos \theta \cos i)N] \\ \dot{h}_y &= rT(-\cos \Omega \sin i) - rN(-\sin \Omega \sin \theta + \cos \Omega \cos \theta \cos i) \\ &= r[-(\cos \Omega \sin i)T + (\sin \Omega \sin \theta - \cos \Omega \cos \theta \cos i)N]\end{aligned}$$

Substitute these expressions into the longitude rate equation, and use

$$h_x^2 + h_y^2 = h^2 \sin^2 i$$

to obtain the following

$$\begin{aligned}\dot{\Omega} &= \frac{rh\{s\Omega si[-(c\Omega si)T + (s\Omega s\theta - c\Omega c\theta ci)N] + c\Omega si[(s\Omega si)T + (c\Omega s\theta + s\Omega c\theta ci)N]\}}{h^2 s^2 i} \\ &= \frac{r}{hsi}[-s\Omega c\Omega siT + (s^2 \Omega s\theta - s\Omega c\Omega c\theta ci)N + s\Omega c\Omega siT + (c^2 \Omega s\theta + s\Omega c\Omega c\theta ci)N] \\ &= \frac{r}{hsi}[(s^2 \Omega s\theta)N + (c^2 \Omega s\theta)N] \\ &= \frac{r \sin \theta}{h \sin i} N\end{aligned}$$

Or

$$\dot{\Omega} = \frac{h \sin \theta}{\mu(1 + e \cos f) \sin i} N = \left[ \frac{a(1 - e^2)}{\mu} \right]^{1/2} \frac{\sin \theta}{(1 + e \cos f) \sin i} N$$

Now consider the specific forces. The gravitational potential, including the  $J_2$  term is

$$\begin{aligned}U_{J_2} &= -\frac{\mu}{r} \left( \frac{r_e}{r} \right)^2 J_2 P_2(\cos \varphi) \\ &= -\frac{\mu}{r} \left( \frac{r_e}{r} \right)^2 J_2 \frac{1}{2} (2 - 3 \sin^2 \varphi)\end{aligned}$$

Define the orbital latitude as

$$\sin^2 \varphi = \frac{x^2 + y^2}{r^2} = \frac{r^2 - z^2}{r^2} = 1 - \frac{z^2}{r^2}$$

Then obtain the following for the preceding potential function:

$$\begin{aligned} U_{J_2} &= -\frac{\mu}{r} \left( \frac{r_e}{r} \right)^2 J_2 \frac{1}{2} \left[ 2 - 3 \left( 1 - \frac{z^2}{r^2} \right) \right] \\ &= -\frac{\mu r_e^2}{2} J_2 \frac{1}{r^3} \left( \frac{z^2}{r^2} - 1 \right) \end{aligned}$$

Operate on this potential with the gradient to yield the specific forces resolved into radial and vertical components as

$$\begin{aligned} f_{J_2} &= \nabla U_{J_2} = \frac{\partial U}{\partial r} \hat{e}_R + \frac{\partial U}{\partial z} \hat{e}_z \\ &= -\frac{\mu r_e^2}{2} J_2 \left[ \left( -3 \frac{1}{r^4} \right) \left( 3 \frac{z^2}{r^2} - 1 \right) + \frac{1}{r^3} \left( -6 \frac{z^2}{r^3} \right) \right] \hat{e}_R - \frac{\mu r_e^2}{2} J_2 \left[ 6 \frac{z}{r^2} \right] \hat{e}_z \\ &= -\mu J_2 r_e^2 \left[ \left( \frac{3}{2r^4} - \frac{15z^2}{2r^6} \right) \hat{e}_R + \left( \frac{3z}{r^5} \right) \hat{e}_z \right] \end{aligned}$$

Express the vertical unit vector in terms of the radial, tangential, and normal vectors as

$$\hat{e}_z = \sin i \sin \theta \hat{e}_R + \sin i \cos \theta \hat{e}_T + \cos i \hat{e}_N$$

and use the following for  $z$ :

$$z = r \cos \varphi = r \sin i \sin \theta$$

to obtain the following expression for the specific force:

$$f_{J_2} = -\mu J_2 r_e^2 \left[ \left( \frac{3}{2r^4} - \frac{15r^2 s^2 i s^2 \theta}{2r^6} \right) \hat{e}_R + \left( \frac{3rsi s\theta}{r^5} \right) (si s\theta \hat{e}_R + sic \theta \hat{e}_T + ci \hat{e}_N) \right]$$

Or

$$f_{J_2} = -\frac{3\mu J_2 r_e^2}{r^4} \left[ \left( \frac{1}{2} - \frac{3s^2 i s^2 \theta}{2} \right) \hat{e}_R + s^2 i s\theta c\theta \hat{e}_T + si ci s\theta \hat{e}_N \right]$$

Return to the longitude rate equation, and use the normal component of specific force, to obtain the following for the longitude rate:

$$\begin{aligned} \dot{\Omega} &= \frac{h}{\mu (1 + e \cos f) \sin i} \left[ -\frac{3\mu J_2 r_e^2}{r^4} \sin i \cos i \sin (\omega + f) \right] \\ &= -\frac{3J_2 r_e^2 h \cos i}{p^4} \sin^2 (\omega + f) (1 + e \cos f)^3 \end{aligned}$$

This rate is continuously varying along the orbit. Of interest is the mean rate of change over an orbital period. Form the mean as

$$\overline{\frac{d\Omega}{dt}} = \frac{1}{2\pi} \int_0^{2\pi} \frac{d\Omega}{dt} dM$$

where the differential mean and true anomaly are

$$dM = ndt$$

$$df = \frac{h}{r^2} dt$$

Change the integration variable as

$$dM = \frac{nr^2}{h} df = \frac{np^2}{h(1+e\cos f)^2} df$$

and substitute this into the preceding integral and evaluate over an orbital period as

$$\begin{aligned} \overline{\frac{d\Omega}{dt}} &= -\frac{3J_2 r_e^2 h \cos i}{p^4} \frac{1}{2\pi} \int_0^{2\pi} \sin^2(\omega + f) (1 + e \cos f)^3 \frac{np^2}{h(1 + e \cos f)^2} df \\ &= -\frac{3J_2 r_e^2 n \cos i}{p^2} \frac{1}{2\pi} \int_0^{2\pi} \sin^2(\omega + f) (1 + e \cos f) df \end{aligned}$$

Or

$$\overline{\frac{d\Omega}{dt}} = -\frac{3}{2} J_2 \left(\frac{r_e}{p}\right)^2 n \cos i.$$

## 8

# Kalman Filtering

The Kalman filter algorithm is used extensively in integrated navigation systems. The algorithm integrates independent redundant sources of navigation information with a reference navigation solution to obtain an optimal estimate of navigation states—position, velocity, and attitude—and other variables that contribute to navigation solution error.

In this chapter, the Kalman filter algorithm is derived in incremental stages and extended for applications. This development and its extensions proceed as follows: recursive weighted least squares for constant systems; recursive weighted least squares for dynamic systems; discrete linear minimum variance estimator;  $U-D$  factored form; summed measurements; combined estimate from two (or more) Kalman filters; and derivative free estimation.

Following the approach of Sage [2], the minimization of a weighted least-squares (WLS) cost function provides an estimator for a system of constants. This result is reformulated into a recursive (RWLS) estimator, again, for a constant system. Structurally, this estimator has the same form as the measurement update portion of the Kalman filter algorithm. The constant system is modified to incorporate dynamics associated with a linear time-invariant system. This model is incorporated into the cost function structure used for the constant system allowing the same minimization approach to be used. This final least-squares step results in a linear discrete estimation algorithm form.

With the form of an optimal linear estimator established, the algorithm is derived again by establishing a minimum variance linear estimator. The linear estimator form is assumed; however, in this derivation, the system model is expanded to include random dynamic model error. The resulting estimator is the linear discrete Kalman filter algorithm. The two approaches, least-squares and minimum variance estimators, result in the same algorithm form.

The standard Kalman filter algorithm has been shown to exhibit poor numerical accuracy for ill-conditioned measurements. This problem is mitigated with the  $U-D$  factored form developed by Bierman [3]. This form of the algorithm is developed for measurement update equations. The  $U-D$  measurement update algorithm has been applied by itself in current integrated navigation systems, not accompanied with the time update portion of the  $U-D$  factored algorithm.

Additional topics are presented next. In an attempt to reduce the effect of noisy measurements, some system integrators sum measurements prior to their being

processed in the Kalman filter algorithm. Modifications to the algorithm modifications to incorporate this pre-processing stage are presented. If outputs from more than one Kalman filter are available, providing estimates of the same or linearly related variables, then individual filters' estimates can be optimally combined. A method for combining filter estimates is presented.

Recent advances in nonlinear estimation have yielded a filtering methodology that does not require the system and measurement linearizations that make up much of this book. Two of these algorithms, unscented Kalman filter (UKF) and divided difference Kalman filter (DDKF), are presented in the final section of this chapter.

Problems that expand upon the material presented are included.

## 8.1 Recursive Weighted Least Squares: Constant Systems

### 8.1.1 Weighted Least Squares

The least-squares estimation algorithm for constant nondynamic systems was presented in Example 2.3. In Problem 2.4, this algorithm was extended by modifying the cost function to include a symmetric weighting matrix  $W$  forming the WLS cost function as

$$J_{\text{WLS}} = (\mathbf{z} - \mathbf{Hx})^T W (\mathbf{z} - \mathbf{Hx}) \quad (8.1)$$

Minimizing this cost function yielded the estimator for the vector  $\mathbf{x}$

$$\hat{\mathbf{x}} = (H^T W H)^{-1} H^T W \mathbf{z} \quad (8.2)$$

If the measurement errors are zero mean random processes, such that  $E[\mathbf{y}] \equiv 0$  and  $E[\mathbf{v}\mathbf{v}^T] \equiv R$ , then the estimation error covariance becomes, replacing the weighting matrix  $W$  with the inverse of the measurement error covariance matrix  $R^{-1}$ ,

$$E[(\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T] \equiv P = (H^T R^{-1} H)^{-1}$$

### 8.1.2 Recursive Weighted Least Squares

If additional data are to be processed, it is desirable to not recompute the estimate using the estimator in Eq. (8.2). It is more efficient to use previously computed estimates in an estimator form that can incorporate an addition data point. To accomplish this, a recursive form of the estimator is to be developed. Consider again the measurement model from Example 2.3.

$$\mathbf{z} = \mathbf{Hx} + \mathbf{v}$$

Expanding this equation in component form yields

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{m1} & h_{m2} & \cdots & h_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_m \end{bmatrix} \quad (8.3)$$

Consider the estimate in Eq. (8.2), as based on  $m$  data points or measurements and designating this estimate as  $\hat{\mathbf{x}}_m$ . With a single additional measurement  $z_{m+1}$ , the measurement vector is modified to include this measurement as

$$\begin{bmatrix} \mathbf{z} \\ z_{m+1} \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{h}^T \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{v} \\ v_{m+1} \end{bmatrix} \quad (8.4)$$

The new estimate of  $\mathbf{x}$ , based on incorporating a single new data point, is specified to be of the form

$$\hat{\mathbf{x}}_{m+1} \equiv \hat{\mathbf{x}}_m + \Delta\mathbf{x} \quad (8.5)$$

where the correction  $\Delta\mathbf{x}$  to the current estimate is to be obtained from the next single measurement processed.

The cost function in Eq. (8.1) is modified again as

$$J_{\text{RWLS}} = \left( \begin{bmatrix} \mathbf{z} \\ z_{m+1} \end{bmatrix} - \begin{bmatrix} \mathbf{H} \\ \mathbf{h}^T \end{bmatrix} \mathbf{x}_{m+1} \right)^T \begin{bmatrix} R^{-1} & 0 \\ 0^T & r^{-1} \end{bmatrix} \left( \begin{bmatrix} \mathbf{z} \\ z_{m+1} \end{bmatrix} - \begin{bmatrix} \mathbf{H} \\ \mathbf{h}^T \end{bmatrix} \mathbf{x}_{m+1} \right) \quad (8.6)$$

In partitioning the inverse of the measurement error covariance matrix with zeros in the last row and column, except for the additional measurement's error variance, it is assumed that the error associated with the additional measurement is independent (uncorrelated) with previous measurement errors. (See later discussions on uncorrelated errors and Table 8.1.) For notational convenience, the following temporary definitions are used:

$$\xi = \begin{bmatrix} \mathbf{z} \\ z_{m+1} \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{H} \\ \mathbf{h}^T \end{bmatrix}, \quad \zeta = \mathbf{x}_{m+1} \quad \text{and} \quad \Lambda = \begin{bmatrix} R & 0 \\ 0 & r \end{bmatrix}$$

In terms of these temporary variable definitions, the cost function in Eq. (8.6) is rewritten as

$$J_{\text{RWLS}} = (\xi - \mathbf{H}\zeta)^T \Lambda (\xi - \mathbf{H}\zeta)$$

In this form, earlier minimizations give the form of the estimator without the need to take derivatives. The result is

$$\hat{\zeta} = (\mathbf{H}^T \Lambda^{-1} \mathbf{H})^{-1} \mathbf{H} \Lambda^{-1} \xi$$

Returning to the original definitions allows this estimate to be written as

$$\hat{\zeta} = \left( \begin{bmatrix} \mathbf{H} \\ \mathbf{h}^T \end{bmatrix}^T \begin{bmatrix} R^{-1} & 0 \\ 0^T & r^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{H} \\ \mathbf{h}^T \end{bmatrix} \right)^{-1} \begin{bmatrix} \mathbf{H} \\ \mathbf{h}^T \end{bmatrix}^T \begin{bmatrix} R^{-1} & 0 \\ 0^T & r^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ z_{m+1} \end{bmatrix}$$

**Table 8.1 Linear discrete Kalman filtering equations**

System model	$\mathbf{x}_{k+1} = \Phi_{k+1,k} \mathbf{x}_k + \Gamma_k \mathbf{w}_k$	$E[\mathbf{w}_i \mathbf{w}_j^T] = [0] \forall i \neq j$ $E[\mathbf{w}_i \mathbf{w}_j^T] = Q_k \forall i = j = k$
Measurement model	$\mathbf{z}_{k+1} = H_{k+1} \mathbf{x}_{k+1} + \mathbf{v}_{k+1}$	$E[\mathbf{v}_i \mathbf{v}_j^T] = [0] \forall i \neq j$ $E[\mathbf{v}_i \mathbf{v}_j^T] = R_k \forall i = j = k$
Initial conditions	$E[\mathbf{x}(t=0)] = \hat{\mathbf{x}}_0$	$E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T] = P_0$
Assumptions	$E[\mathbf{w}_i \mathbf{v}_j^T] = [0] \forall i, j$	{considered here}
Propagate stage	$\tilde{\mathbf{x}}_{k+1} = \Phi_{k+1,k} \hat{\mathbf{x}}_k$	—
Propagate covariance	$\tilde{P}_{k+1} = \Phi_{k+1,k} \hat{P}_k \Phi_{k+1,k}^T + \Gamma_k Q_k \Gamma_k^T$	—
Measurement update	$\begin{aligned} \hat{\mathbf{x}}_{k+1} &= \tilde{\mathbf{x}}_{k+1} + K_{k+1} \\ &\quad \times [\mathbf{z}_{k+1} - H_{k+1} \tilde{\mathbf{x}}_{k+1}] \\ \hat{P}_{k+1} &= [I - K_{k+1} H_{k+1}] \tilde{P}_{k+1} \\ K_{k+1} &= \tilde{P}_{k+1} H_{k+1}^T \\ &\quad \times (H_{k+1} \tilde{P}_{k+1} H_{k+1}^T + R_{k+1})^{-1} \end{aligned}$	—

or, in expanded form,

$$\hat{\xi} = (H^T R^{-1} H + \mathbf{h} r^{-1} \mathbf{h}^T)^{-1} [H^T R^{-1} \mathbf{z} + \mathbf{h} r^{-1} z_{m+1}] \quad (8.7)$$

Before proceeding with the estimator development, a recursive form for the *error covariance matrix*, the uncertainty in the estimate, is developed in the following. Recall from Problem 2.4 that the error covariance matrix, processing  $m$  data points, resulting from the WLS estimator is

$$P_m^{-1} = H^T R^{-1} H$$

then using the temporary variable definitions and expanding terms yield

$$\begin{aligned} P_{m+1}^{-1} &= H^T \Lambda^{-1} H \\ &= H^T R^{-1} H + \mathbf{h} r^{-1} \mathbf{h}^T \\ &\equiv P_m^{-1} + \mathbf{h} r^{-1} \mathbf{h}^T \end{aligned} \quad (8.8)$$

The matrix inversion lemma given in Example 2.2 is used to express the error covariance matrix as itself rather than its inverse. Using this lemma, the following

form for the error covariance matrix is obtained:

$$P_{m+1} = P_m - P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T P_m \quad (8.9)$$

or grouping terms in this equation that reoccur later and writing it as

$$P_{m+1} = [I - \mathbf{k} \mathbf{h}^T] P_m \quad (8.10)$$

where the grouped terms form the vector  $\mathbf{k}$  as

$$\mathbf{k}_m = P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \quad (8.11)$$

Returning to the *estimator* in Eq. (8.7), first using the expression for  $P_{m+1}$  in the second line in Eq. (8.8) and then Eq. (8.9), the estimate is written as

$$\begin{aligned} \hat{\mathbf{x}}_{m+1} &= P_{m+1} [H^T R^{-1} \mathbf{z} + \mathbf{h} r^{-1} z_{m+1}] \\ &= P_m H^T R^{-1} \mathbf{z} + P_m \mathbf{h} r^{-1} z_{m+1} \\ &\quad - P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T P_m [H^T R^{-1} \mathbf{z} + \mathbf{h} r^{-1} z_{m+1}] \end{aligned}$$

Recognizing terms in this expression as the estimate from the preceding step,

$$\hat{\mathbf{x}}_m = P_m H^T R^{-1} \mathbf{z} = (H^T R^{-1} H)^{-1} H^T R^{-1} \mathbf{z}$$

After grouping terms, this equation can be rewritten as

$$\begin{aligned} \hat{\mathbf{x}}_{m+1} &= \hat{\mathbf{x}}_m + \left[ P_m - P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T P_m \right] [\mathbf{h} r^{-1} z_{m+1}] \\ &\quad - P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T \hat{\mathbf{x}}_m \end{aligned}$$

Factoring  $P_m$  from the left, this equation becomes

$$\begin{aligned} \hat{\mathbf{x}}_{m+1} &= \hat{\mathbf{x}}_m + P_m \left[ I - \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T P_m \right] [\mathbf{h} r^{-1} z_{m+1}] \\ &\quad - P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T \hat{\mathbf{x}}_m \end{aligned}$$

Multiplying the square bracket term from the right by  $\mathbf{h}$  and factoring  $\mathbf{h}$  out of this bracket from the left yields

$$\begin{aligned} \hat{\mathbf{x}}_{m+1} &= \hat{\mathbf{x}}_m + P_m \mathbf{h} \left[ 1 - (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T P_m \right] \mathbf{h} r^{-1} z_{m+1} \\ &\quad - P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \mathbf{h}^T \hat{\mathbf{x}}_m \end{aligned}$$

The terms in the square brackets become scalars and are simplified as

$$\left[ 1 - (\mathbf{h}^T \mathbf{P}_m \mathbf{h} + r)^{-1} \mathbf{h}^T \mathbf{P}_m \mathbf{h} \right] = 1 - \frac{\mathbf{h}^T \mathbf{P}_m \mathbf{h}}{\mathbf{h}^T \mathbf{P}_m \mathbf{h} + r} = \frac{r}{\mathbf{h}^T \mathbf{P}_m \mathbf{h} + r}$$

Using this result, the estimate equation is rewritten as

$$\hat{\mathbf{x}}_{m+1} = \hat{\mathbf{x}}_m + P_m \mathbf{h} (\mathbf{h}^T \mathbf{P}_m \mathbf{h} + r)^{-1} z_{m+1} - P_m \mathbf{h} (\mathbf{h}^T \mathbf{P} \mathbf{h} + r)^{-1} \mathbf{h}^T \hat{\mathbf{x}}_m$$

Using the definition of the gain  $\mathbf{k}_m$  from Eq. (8.11), the estimate equation becomes

$$\hat{\mathbf{x}}_{m+1} = \hat{\mathbf{x}}_m + \mathbf{k}_m [z_{m+1} - \mathbf{h}^T \hat{\mathbf{x}}_m] \quad (8.12)$$

Equations (8.10–8.12) form the recursive weighted least-squares (RWLS) estimator and are functionally similar to the measurement update equations of the Kalman filter algorithm. This development has assumed a constant nondynamic state vector  $\mathbf{x}$ .

These equations were obtained for a single new measurement update. This form also holds for vector updates. For the vector update case, the measurement  $z_{m+1}$  and vector  $\mathbf{h}$  become  $\mathbf{z}_{m+1}$  and  $\mathbf{H}_{m+1}$ , respectively. The equations also are valid for measurement step depend variances, that is,  $r_m$ , and forms of the measurement vector represented by  $\mathbf{h}_m$ .

### Example 8.1 Data Trends

If the state vector is a scalar constant, the covariance update equation (8.9) can be written as

$$p_{m+1} = \left[ 1 - \frac{p_m h^2}{(p_m h^2 + r)} \right] p_m$$

Because both the measurement variance and the error variance are positive, the quantity in the bracket is less than one, and the new error variance is always less than the previous variance. This leads to the continual reduction of the estimation error with each new measurement until the error variance is reduced to an arbitrarily small value  $\sim 0$ . This would reflect improved knowledge of the state's value by its estimate. With small error variance, the gain  $k$  would also be small. New values of the state produced from Eq. (8.12) would change little from previous values. This is related to the divergence problem in some applications of Kalman filtering.

In this example, suppose that the measurement variance is very large. This corresponds to large measurement error magnitudes compared to the states to be estimated. From the preceding equation, it is seen that the updated error variance would also be changed little from the earlier value. Because the gain  $\mathbf{k}$  can also be expressed as (see Problem 8.1)

$$\mathbf{k}_m = P_{m+1} \mathbf{h} r^{-1}$$

the correction of the prior estimate value would also be very small. New values from the state update in Eq. (8.12) would be unchanged. Therefore, for the case of large measurement variances, the estimation error and state would remain unchanged from their initial values.

## 8.2 Recursive Weighted Least Squares: Dynamic Systems

A dynamic system is in general nonlinear as discussed in Chapter 2. This system can also be driven or disturbed by additive error, which in most applications in navigation systems is the result of unmodeled dynamical elements. The applications of Kalman filtering in integrated navigation systems assume a linearized form of state dynamics. This linearized form is used next in the development of a recursive weighted least-squares estimator for a dynamic system. At this point, it is assumed that there are no additive errors to the dynamics. This approach will continue the preceding process by using the same cost function structure. Using this approach, the dynamics are incorporated into the measurement matrix resulting in a time-dependent measurement, and, in effect, a constant state vector as assumed in the preceding.

The linear state dynamics, with no additive disturbances, is represented by the following discrete-time equation:

$$\mathbf{x}_{i+1} = \Phi_{i+1,i} \mathbf{x}_i \quad (8.13)$$

where the state is propagated in time, from the previous time instant  $i$  to the next instant  $i + 1$ , by the state transition matrix  $\Phi_{i+1,i}$ . The state is observed by measurements at the instant  $i$  by

$$\mathbf{z}_i = H_i \mathbf{x}_i + \mathbf{v}_i \quad (8.14)$$

where the allowance for time dependence of the measurement, represented by the  $H_i$  matrix, and measurement error  $\mathbf{v}_i$ , are indicated. A sequence of measurements, up to the  $k$ th time instant, is related to the state vector at the  $k$ th time instant by incorporating into the measurement matrix at each time instant the state transition matrix to propagate the state from that time instant to the  $k$ th time instant

$$\begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_k \end{bmatrix} = \begin{bmatrix} H_1 \Phi_{1,k} \\ H_2 \Phi_{2,k} \\ \vdots \\ H_k \Phi_{k,k} \end{bmatrix} \mathbf{x}_k + \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_k \end{bmatrix} \quad (8.15)$$

Writing this in a composite form as

$$\boldsymbol{\xi}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \quad (8.16)$$

Recalling the WLS cost function [see Eq. (8.1)]

$$J_{\text{WLS}} = (\boldsymbol{\xi}_k - \mathbf{H}_k \mathbf{x})^T \Lambda_k^{-1} (\boldsymbol{\xi}_k - \mathbf{H}_k \mathbf{x}) \quad (8.17)$$

and minimizing with respect to  $\mathbf{x}_k$  [see Eq. (8.2)] yield

$$\hat{\mathbf{x}}_k = (\mathbf{H}_k^T \Lambda_k^{-1} \mathbf{H}_k)^{-1} \mathbf{H}_k^T \Lambda_k^{-1} \boldsymbol{\xi}_k \quad (8.18)$$

As with RWLS for constant systems, consider an additional vector measurement at time instant  $k + 1$ :

$$\mathbf{z}_{k+1} = \mathbf{H}_{k+1} \mathbf{x}_{k+1} + \mathbf{v}_{k+1} \quad (8.19)$$

The measurement represented by Eq. (8.16) is modified to include this additional vector measurement at time instant  $k + 1$  as

$$\begin{bmatrix} \boldsymbol{\xi}_k \\ \mathbf{z}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_k \Phi_{k,k+1} \\ \mathbf{H}_{k+1} \end{bmatrix} \mathbf{x}_{k+1} + \begin{bmatrix} \mathbf{v}_k \\ \mathbf{v}_{k+1} \end{bmatrix} \quad (8.20)$$

The modified cost function becomes

$$\begin{aligned} J_{\text{RLWS}} &= \left( \begin{bmatrix} \boldsymbol{\xi}_k \\ \mathbf{z}_{k+1} \end{bmatrix} - \begin{bmatrix} \mathbf{H}_k \Phi_{k,k+1} \\ \mathbf{H}_{k+1} \end{bmatrix} \mathbf{x}_{k+1} \right)^T \\ &\quad \times \begin{bmatrix} \Lambda_k^{-1} & [0] \\ [0] & R_{k+1}^{-1} \end{bmatrix} \left( \begin{bmatrix} \boldsymbol{\xi}_k \\ \mathbf{z}_{k+1} \end{bmatrix} - \begin{bmatrix} \mathbf{H}_k \Phi_{k,k+1} \\ \mathbf{H}_{k+1} \end{bmatrix} \mathbf{x}_{k+1} \right) \\ &= (\boldsymbol{\xi}_k - \mathbf{H}_k \Phi_{k,k+1} \mathbf{x}_{k+1})^T \Lambda_k^{-1} (\boldsymbol{\xi}_k - \mathbf{H}_k \Phi_{k,k+1} \mathbf{x}_{k+1}) \\ &\quad + (\mathbf{z}_{k+1} - \mathbf{H}_{k+1} \mathbf{x}_{k+1})^T R_{k+1}^{-1} (\mathbf{z}_{k+1} - \mathbf{H}_{k+1} \mathbf{x}_{k+1}) \end{aligned} \quad (8.21)$$

The new estimate of  $\mathbf{x}$ , based on incorporating the new vector measurement, is specified to be the sum of the propagated earlier time-step estimate and an incremental change as a result of the new vector measurement. This is expressed as

$$\hat{\mathbf{x}}_{k+1} = \Phi_{k+1,k} \hat{\mathbf{x}}_k + \Delta \hat{\mathbf{x}}_{k+1} \quad (8.22)$$

which implies

$$\Phi_{k,k+1} \hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_k + \Phi_{k,k+1} \Delta \hat{\mathbf{x}}_{k+1}$$

Using these relationships, the cost function in Eq. (8.21) becomes (momentarily dropping the  $\hat{\cdot}$ )

$$\begin{aligned}
 J_{\text{RLWS}} &= [\xi_k - H_k(\mathbf{x}_k + \Phi_{k,k+1}\Delta\mathbf{x}_{k+1})]^T \Lambda_k^{-1} [\xi_k - H_k(\mathbf{x}_k + \Phi_{k,k+1}\Delta\mathbf{x}_{k+1})] \\
 &\quad + [z_{k+1} - H_{k+1}(\Phi_{k+1,k}\mathbf{x}_k + \Delta\mathbf{x}_{k+1})]^T \\
 &\quad \times R_{k+1}^{-1} [z_{k+1} - H_{k+1}(\Phi_{k+1,k}\mathbf{x}_k + \Delta\mathbf{x}_{k+1})] \\
 &= \xi_k^T \Lambda_k^{-1} \xi_k - \mathbf{x}_k^T H_k^T \Lambda_k^{-1} \xi_k - \Delta\mathbf{x}_{k+1}^T \Phi_{k,k+1}^T H_k^T \Lambda_k^{-1} \xi_k - \xi_k^T \Lambda_k^{-1} H_k \mathbf{x}_k \\
 &\quad - \xi_k^T \Lambda_k^{-1} H_k \Phi_{k,k+1} \Delta\mathbf{x}_{k+1} + \mathbf{x}_k^T H_k^T \Lambda_k^{-1} H_k \mathbf{x}_k \\
 &\quad + \mathbf{x}_k^T H_k^T \Lambda_k^{-1} H_k \Phi_{k,k+1} \Delta\mathbf{x}_{k+1} + \Delta\mathbf{x}_{k+1}^T \Phi_{k,k+1}^T H_k^T \Lambda_k^{-1} H_k \mathbf{x}_k \\
 &\quad + \Delta\mathbf{x}_{k+1}^T \Phi_{k,k+1}^T H_k^T \Lambda_k^{-1} H_k \Phi_{k,k+1} \Delta\mathbf{x}_{k+1} \\
 &\quad + [z_{k+1} - H_{k+1}(\Phi_{k+1,k}\mathbf{x}_k + \Delta\mathbf{x}_{k+1})]^T \\
 &\quad \times R_{k+1}^{-1} [z_{k+1} - H_{k+1}(\Phi_{k+1,k}\mathbf{x}_k + \Delta\mathbf{x}_{k+1})]
 \end{aligned} \tag{8.23}$$

From Eq. (8.18)

$$\mathbf{x}_k^T H_k^T \Lambda_k^{-1} H_k = \xi_k^T \Lambda_k^{-1} H_k \tag{8.24}$$

Using Eq. (8.24), several terms in Eq. (8.23) cancel, leaving

$$\begin{aligned}
 J_{\text{RLWS}} &= \Delta\mathbf{x}_{k+1}^T \Phi_{k,k+1}^T H_k^T \Lambda_k^{-1} H_k \Phi_{k,k+1} \Delta\mathbf{x}_{k+1} \\
 &\quad + [z_{k+1} - H_{k+1}(\Phi_{k+1,k}\mathbf{x}_k + \Delta\mathbf{x}_{k+1})]^T \\
 &\quad \times R_{k+1}^{-1} [z_{k+1} - H_{k+1}(\Phi_{k+1,k}\mathbf{x}_k + \Delta\mathbf{x}_{k+1})]
 \end{aligned} \tag{8.25}$$

Minimizing  $J_{\text{RWLS}}$  with respect to  $\Delta\mathbf{x}_{k+1}$  yields

$$\begin{aligned}
 \Delta\hat{\mathbf{x}}_{k+1} &= (\Phi_{k,k+1}^T H_k^T \Lambda_k^{-1} H_k \Phi_{k,k+1} + H_{k+1}^T R_{k+1}^{-1} H_{k+1})^{-1} \\
 &\quad \times H_{k+1}^T R_{k+1}^{-1} [z_{k+1} - H_{k+1} \Phi_{k+1,k} \hat{\mathbf{x}}_k]
 \end{aligned} \tag{8.26}$$

As before, defining

$$P_k^{-1} \equiv H_k^T \Lambda_k^{-1} H_k \tag{8.27}$$

and

$$P_{k+1}^{-1} = \Phi_{k,k+1}^T P_k^{-1} \Phi_{k+1,k} + H_{k+1}^T R_{k+1}^{-1} H_{k+1} \tag{8.28}$$

allows the incremental state vector estimate, given in Eq. (8.26), to be written as

$$\Delta \hat{\mathbf{x}}_{k+1} = P_{k+1} H_{k+1}^T R_{k+1}^{-1} [\mathbf{z}_{k+1} - H_{k+1} \Phi_{k+1,k} \hat{\mathbf{x}}_k] \quad (8.29)$$

Substituting this equation into Eq. (8.22) yields for the state vector estimate at time instant  $k + 1$

$$\hat{\mathbf{x}}_{k+1} = \Phi_{k+1,k} \hat{\mathbf{x}}_k + P_{k+1} H_{k+1}^T R_{k+1}^{-1} [\mathbf{z}_{k+1} - H_{k+1} \Phi_{k+1,k} \hat{\mathbf{x}}_k] \quad (8.30)$$

Finally, using the matrix inversion lemma for Eq. (8.28) yields the update for the error covariance matrix

$$\begin{aligned} P_{k+1} &= \Phi_{k+1,k} P_k \Phi_{k+1,k}^T \\ &\quad - \Phi_{k+1,k} P_k \Phi_{k+1,k}^T H_{k+1}^T (H_{k+1} \Phi_{k+1,k} P_k \Phi_{k+1,k}^T H_{k+1}^T + R_{k+1})^{-1} \\ &\quad \times H_{k+1} \Phi_{k+1,k} P_k \Phi_{k+1,k}^T \end{aligned} \quad (8.31)$$

This equation can be simplified by adopting the nomenclature of the caret as the estimate after a measurement update and the tilde as the estimate after a time update as

$$\tilde{P}_{k+1} \equiv \Phi_{k+1,k} \hat{P}_k \Phi_{k+1,k}^T \quad (8.32)$$

then Eq. (8.31) becomes

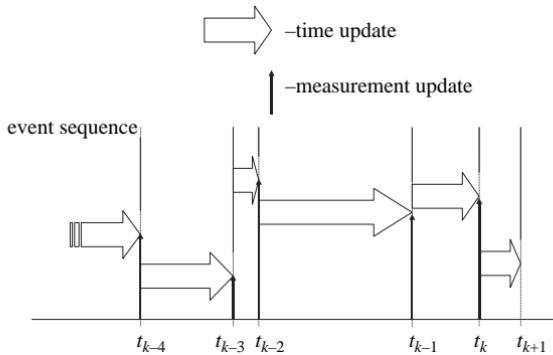
$$\hat{P}_{k+1} = \tilde{P}_{k+1} - \tilde{P}_{k+1} H_{k+1}^T (H_{k+1} \tilde{P}_{k+1} H_{k+1}^T + R_{k+1})^{-1} H_{k+1} \tilde{P}_{k+1} \quad (8.33)$$

The model of the process given in Eq. (8.13) assumes no disturbing noise in state dynamics. If system dynamics are governed by the following discrete-time equation

$$\mathbf{x}_{k+1} = \Phi_{k+1,k} \mathbf{x}_k + \Gamma_k \mathbf{w}_k \quad (8.34)$$

the resulting estimation algorithm is summarized in Table 8.1. Equations in this table are the standard Kalman filter form for discrete-time measurement update form.

The filter algorithm processing flow is illustrated in Fig. 8.1. Beginning at some prior time instant, that is,  $k - 5$ , the state vector and covariance matrix are propagated in time (time update—open arrow) to the  $k - 4$  time instant, at which measurements are available for use in updating the state and covariance matrix. At this  $k - 4$  time instant, one or more measurements can be processed by the state vector and covariance matrix measurement update equations (measurement update—vertical arrow). With the measurement update completed, the state vector and covariance matrix are again propagated in time (time update—open arrow) to the  $k - 3$  time, at which measurements are



**Fig. 8.1** Kalman filter algorithm processing flow.

available for updating the state and covariance matrix. This alternating event sequence of time and measurement updates is repeated until stopped. Figure 8.1 shows irregular time intervals and variable numbers of measurements allowed by the algorithm.

Examining the equations in Table 8.1, certain observations are useful. If the system model, measurement model, and disturbance error characteristics are linear and constant with time, then the covariance matrix does not depend on the states. For this constant system, the covariance matrix propagation and updates can be performed offline. Once computed, the gain matrix  $K$  can be stored as a function of time within the onboard computer. This approach is termed the *scheduled gain* filter implementation. As such, it is not a dynamic Kalman filter but a recursive estimator. Applications of this approach have been used for inertial navigation system ground alignment and integrated navigation systems for quasi-steady dynamic conditions.

### 8.3 Discrete Linear Minimum Variance Estimator

Linear discrete Kalman filter equations are derived in the following by a different approach: that of minimizing the estimation error variance. The objective of the derivation is to obtain a linear unbiased estimator for the state vector  $\mathbf{x}$ .

Consider again the dynamical process described by Eq. (8.34):

$$\mathbf{x}_{k+1} = \Phi_{k+1,k} \mathbf{x}_k + \Gamma_k \mathbf{w}_k$$

where the noise  $\mathbf{w}_k$  is a zero-mean uncorrelated process. A conditional expectation is considered as an operator whose result is based on whether the object of the operator, that is,  $\mathbf{e}_k$  has zero mean or is correlated. Assume the estimate (expectation) of this process can be propagated in time by

$$\tilde{\mathbf{x}}_{k+1} \equiv \Phi_{k+1,k} \hat{\mathbf{x}}_k \quad (8.35)$$

The state estimate's error, from the measurement update, is defined as

$$\hat{\boldsymbol{\epsilon}}_k = \hat{\mathbf{x}}_k - \mathbf{x}_k \quad (8.36)$$

The error at the next time step  $k + 1$  is obtained from these equations as

$$\begin{aligned} \tilde{\boldsymbol{\epsilon}}_{k+1} &= \tilde{\mathbf{x}}_{k+1} - \mathbf{x}_{k+1} \\ &= \Phi_{k+1,k} \hat{\mathbf{x}}_k - (\Phi_{k+1,k} \mathbf{x}_k + \Gamma_k \mathbf{w}_k) \\ &= \Phi_{k+1,k} \hat{\boldsymbol{\epsilon}}_k - \Gamma_k \mathbf{w}_k \end{aligned} \quad (8.37)$$

The estimate/expectation of this error is

$$\begin{aligned} E[\tilde{\boldsymbol{\epsilon}}_{k+1}] &= E[\tilde{\mathbf{x}}_{k+1} - \mathbf{x}_{k+1}] \\ &= \Phi_{k+1,k} E[\hat{\boldsymbol{\epsilon}}_k] - \Gamma_k E[\mathbf{w}_k] \\ &= 0 \end{aligned} \quad (8.38)$$

where it is assumed

$$E[\hat{\boldsymbol{\epsilon}}_k] = 0 \quad (8.39)$$

and

$$E[\mathbf{w}_k] = 0 \quad (8.40)$$

That is, if initially unbiased, the linear system's estimate remains unbiased.

The covariance of the estimation error at the next time step  $k + 1$  is

$$\begin{aligned} \tilde{P}_{k+1} &\equiv E[\tilde{\boldsymbol{\epsilon}}_{k+1} \tilde{\boldsymbol{\epsilon}}_{k+1}^T] \\ &= E\left[(\Phi_{k+1,k} \hat{\boldsymbol{\epsilon}}_k - \Gamma_k \mathbf{w}_k)(\Phi_{k+1,k} \hat{\boldsymbol{\epsilon}}_k - \Gamma_k \mathbf{w}_k)^T\right] \\ &= \Phi_{k+1,k} E[\hat{\boldsymbol{\epsilon}}_k \hat{\boldsymbol{\epsilon}}_k^T] \Phi_{k+1,k}^T - \Gamma_k E[\mathbf{w}_k \mathbf{w}_k^T] \Phi_{k+1,k}^T \\ &\quad - \Phi_{k+1,k} E[\hat{\boldsymbol{\epsilon}}_k \mathbf{w}_k^T] \Gamma_k^T + \Gamma_k E[\mathbf{w}_k \mathbf{w}_k^T] \Gamma_k^T \end{aligned} \quad (8.41)$$

or

$$\tilde{P}_{k+1} = \Phi_{k+1,k} \hat{P}_k \Phi_{k+1,k}^T + \Gamma_k Q_k \Gamma_k^T \quad (8.42)$$

The two middle terms in Eq. (8.41) are assumed zero because

$$\begin{aligned} E[\mathbf{w}_k \hat{\boldsymbol{\epsilon}}_k^T] &= E\left[\mathbf{w}_k (\Phi_{k,k-1} \hat{\boldsymbol{\epsilon}}_{k-1} - \Gamma_{k-1} \mathbf{w}_{k-1})^T\right] \\ &= E[\mathbf{w}_k \hat{\boldsymbol{\epsilon}}_{k-1}^T] \Phi_{k,k-1}^T - E[\mathbf{w}_k \mathbf{w}_{k-1}^T] \Gamma_k^T \\ &= [0] \end{aligned} \quad (8.43)$$

The first term in this equation is zero because the noise  $\mathbf{w}$  at the current time step  $k$  is independent of the error  $\hat{\boldsymbol{\epsilon}}$  at the preceding time step  $k-1$  (see Problem 8.5), and the second term is zero because it is assumed that the process noise is uncorrelated (see Table 8.1).

Equations (8.35) and (8.42) define the estimate and the error in the estimate, both propagated in time. With insight gained from the preceding section, assume a linear form for the state estimator given by

$$\hat{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k + K_{k+1} [\mathbf{z}_{k+1} - H_{k+1} \tilde{\mathbf{x}}_k] \quad (8.44)$$

and, from the definition of the estimate error in Eq. (8.36), and rewritten as

$$\hat{\mathbf{x}}_k = \mathbf{x}_k + \hat{\boldsymbol{\epsilon}}_k \quad (8.45)$$

Equation (8.44) can be rewritten as

$$\begin{aligned} \mathbf{x}_{k+1} + \hat{\boldsymbol{\epsilon}}_{k+1} &= \Phi_{k+1,k}(\mathbf{x}_k + \hat{\boldsymbol{\epsilon}}_k) + K_{k+1} [\mathbf{z}_{k+1} - H_{k+1} \Phi_{k+1,k}(\mathbf{x}_k + \hat{\boldsymbol{\epsilon}}_k)] \\ &= \Phi_{k+1,k} \mathbf{x}_k + \Phi_{k+1,k} \hat{\boldsymbol{\epsilon}}_k \\ &\quad + K_{k+1} [H_{k+1}(\mathbf{x}_{k+1} - \Phi_{k+1,k} \mathbf{x}_k) + \mathbf{v}_{k+1} - H_{k+1} \Phi_{k+1,k} \hat{\boldsymbol{\epsilon}}_k] \end{aligned} \quad (8.46)$$

Using Eq. (8.34), this equation becomes

$$\hat{\boldsymbol{\epsilon}}_{k+1} = [I - K_{k+1} H_{k+1}] \Phi_{k+1,k} \hat{\boldsymbol{\epsilon}}_k - [I - K_{k+1} H_{k+1}] \Gamma_k \mathbf{w}_k + K_{k+1} \mathbf{v}_{k+1} \quad (8.47)$$

The corresponding estimation error covariance matrix is

$$\begin{aligned} \hat{P}_{k+1} &\equiv E[\hat{\boldsymbol{\epsilon}}_{k+1} \hat{\boldsymbol{\epsilon}}_{k+1}^T] \\ &= [I - K_{k+1} H_{k+1}] \Phi_{k+1,k} \hat{P}_k \Phi_{k+1,k}^T [I - K_{k+1} H_{k+1}]^T \\ &\quad + [I - K_{k+1} H_{k+1}] \Gamma_k Q_k \Gamma_k^T [I - K_{k+1} H_{k+1}]^T + K_{k+1} R_{k+1} K_{k+1}^T \end{aligned} \quad (8.48)$$

Using Eq. (8.42), the first and second terms are combined, resulting in

$$\hat{P}_{k+1} = [I - K_{k+1} H_{k+1}] \tilde{P}_{k+1} [I - K_{k+1} H_{k+1}]^T + K_{k+1} R_{k+1} K_{k+1}^T \quad (8.49)$$

Thus far in this development there have been no assumptions concerning the gain matrix  $K_{k+1}$ , other than its use in the linear estimator given in Eq. (8.44).

Therefore, it can be assumed that Eq. (8.49) is true for any gain matrix  $K_{k+1}$ . This form, also known as Joseph's form, has been used in covariance simulations to evaluate reduced state filter implementations with truth models that include more errors than the reduced state Kalman filter implementation.

The gain matrix  $K_{k+1}$  that minimizes Eq. (8.49) yields a minimum variance estimator. Taking the variance of this equation and setting equal to the null matrix yields for the optimal gain

$$K_{k+1} = \tilde{P}_{k+1} H_{k+1}^T (H_{k+1} \tilde{P}_{k+1} H_{k+1}^T + R_{k+1})^{-1} \quad (8.50)$$

This gain is the same as derived earlier using the recursive WLS approach. When this gain matrix is substituted into Eq. (8.49), the earlier form of the error covariance matrix measurement update equation is obtained.

### Example 8.2 From Joseph's Form [Eq. (8.49)] to Standard Kalman Form

Rewrite Eqs. (8.49) and (8.50) without subscripts:

$$\hat{P} = [I - KH]\tilde{P}[I - KH]^T + KRK^T$$

and

$$K = \tilde{P}H^T(H\tilde{P}H^T + R)^{-1}$$

Substituting the second equation into the first, and expanding the measurement update for the error covariance becomes

$$\begin{aligned} \hat{P} &= \tilde{P} - KH\tilde{P} - \tilde{P}H^T K^T + KH\tilde{P}H^T K^T + KRK^T \\ &= \tilde{P} - \tilde{P}H^T(H\tilde{P}H^T + R)^{-1}H\tilde{P} - \tilde{P}H^T(\ )^{-1}H\tilde{P} \\ &\quad + \tilde{P}H^T(\ )^{-1}H\tilde{P}H^T(\ )^{-1}H\tilde{P} + \tilde{P}H^T(\ )^{-1}R(\ )^{-1}H\tilde{P} \\ &= \tilde{P} - \tilde{P}H^T(\ )^{-1}H\tilde{P} - \tilde{P}H^T(\ )^{-1}H\tilde{P} + \tilde{P}H^T(\ )^{-1}(\ )( )^{-1}H\tilde{P} \\ &= \tilde{P} - \tilde{P}H^T(\ )^{-1}H\tilde{P} - \tilde{P}H^T(\ )^{-1}H\tilde{P} + \tilde{P}H^T(\ )^{-1}H\tilde{P} \\ &= \tilde{P} - \tilde{P}H^T(H\tilde{P}H^T + R)^{-1}H\tilde{P} \\ &= [I - KH]\tilde{P} \end{aligned}$$

### 8.4 U-D Factored Form

Equations summarized in Table 8.1 represent one of many forms of the Kalman filtering algorithm. This algorithm has been modified into an *U-D*

factored form by Bierman [3]. The  $U-D$  algorithm provides enhanced numerical precision, reduced storage requirements, and a reduced number of numerical operations. This algorithm processes measurements sequentially (single scalar measurements), thus eliminating the need for matrix inversions. In the following, the nomenclature associated with this algorithm is presented, and, in an example, the form of the algorithm presented by Bierman is developed.

The  $P$  matrix is factored as (see Example 2.1)

$$P = UDU^T \quad (8.51)$$

where the  $U$  matrix is a upper triangular matrix with unity diagonal values and zeros below the diagonal. The  $D$  matrix is a diagonal matrix with zeros off the diagonal.

#### 8.4.1 Measurement Update

The  $P$  matrix, from Table 8.1, can be rewritten for a scalar update as

$$\hat{P} = \tilde{P} - \frac{1}{\alpha} \tilde{P} \mathbf{h} \mathbf{h}^T \tilde{P} \quad (8.52)$$

where, for the scalar update, the measurement residual variance is defined as

$$\alpha = \mathbf{h}^T \tilde{P} \mathbf{h} + r \quad (8.53)$$

Expressing the covariance matrix measurement update in this factored form,

$$\tilde{P} = \tilde{U} \tilde{D} \tilde{U}^T \quad (8.54)$$

Then, substituting Eq. (8.54) into Eq. (8.52) yields

$$\begin{aligned} \hat{P} &= \tilde{U} \tilde{D} \tilde{U}^T - \frac{1}{\alpha} \tilde{U} \tilde{D} \tilde{U}^T \mathbf{h} \mathbf{h}^T \tilde{U} \tilde{D} \tilde{U}^T \\ &= \tilde{U} \left[ \tilde{D} - \frac{1}{\alpha} \tilde{D} \tilde{U}^T \mathbf{h} \mathbf{h}^T \tilde{U} \tilde{D} \right] \tilde{U}^T \end{aligned} \quad (8.55)$$

defining the repeated terms in this equation as

$$\mathbf{f} \equiv \tilde{U}^T \mathbf{h} \quad (8.56)$$

and

$$\mathbf{g} \equiv \tilde{D} \mathbf{f} \quad (8.57)$$

Then, Eq. (8.55) can then be rewritten as

$$\hat{P} = \tilde{U} \left[ \tilde{D} - \frac{1}{\alpha} \mathbf{g} \mathbf{g}^T \right] \tilde{U}^T \quad (8.58)$$

The gain  $K$  in Table 8.1 can be written in terms of the factors as

$$\begin{aligned} \mathbf{k} &= \frac{1}{\alpha} \tilde{\mathbf{P}} \mathbf{h} \\ &= \frac{1}{\alpha} \tilde{\mathbf{U}} \tilde{\mathbf{D}} \tilde{\mathbf{U}}^T \mathbf{h} \end{aligned} \quad (8.59)$$

or, using the definitions in Eqs. (8.56) and (8.57),

$$\mathbf{k} = \frac{1}{\alpha} \tilde{\mathbf{U}} \mathbf{g} \quad (8.60)$$

Using the preceding notation, the  $U-D$  measurement update algorithm is presented next. It is assumed that the measurement variance is nonsingular, that is,  $\alpha \neq 0$ . Initially, form the vectors  $\mathbf{f}$  and  $\mathbf{g}$ , as given in Eqs. (8.56) and (8.57).

$$\mathbf{f} \implies f_i, \quad i = 1, 2, \dots, n \quad \text{and} \quad \mathbf{g} \implies g_i, \quad i = 1, 2, \dots, n$$

$$g_1 = \tilde{D}_1 f_1$$

$$\alpha_1 \equiv r + f_1 g_1$$

$$\hat{D}_1 = \frac{r}{\alpha_1} \tilde{D}_1$$

for  $j = 2, \dots, n$

$$\alpha_j = \alpha_{j-1} + f_j g_j$$

$$\hat{D}_j = \frac{\alpha_{j-1}}{\alpha_j} \tilde{D}_j$$

$$\lambda_j = -\frac{f_j}{\alpha_{j-1}}$$

for  $i = 1, \dots, j-1$

$$\hat{U}_{ij} = \tilde{U}_{ij} + g_i \lambda_j$$

$$g_i = g_{i-1} + g_j \tilde{U}_{ij}$$

$$k_i = \frac{g_i}{\alpha}$$

The algorithm's initialization is accomplished from the initial factorization of the initial covariance matrix  $P_0$ , as in Example 2.1. The estimation covariance matrix  $P$  can be reformed from the updated  $U-D$  factors, as given in Eq. (8.54).

#### 8.4.2 Time Update

This section introduces the nomenclature of the  $U-D$  factored form's time update. The  $P$  matrix time update (propagation) from Table 8.1 can be rewritten as

$$\tilde{P} = \Phi \hat{P} \Phi^T + \Gamma Q \Gamma^T \quad (8.61)$$

or

$$\begin{aligned}\tilde{P} &= \Phi \hat{U} \hat{D} \hat{U}^T \Phi^T + \Gamma Q \Gamma^T \\ &= [\Phi \hat{U} \quad \Gamma] \begin{bmatrix} \hat{D} & [0] \\ [0] & Q \end{bmatrix} \begin{bmatrix} \hat{U}^T \Phi^T \\ \Gamma^T \end{bmatrix} r\end{aligned}\quad (8.62)$$

Identifying the matrices

$$W = [\Phi \hat{U} \quad \Gamma] \quad (8.63)$$

and

$$D = \begin{bmatrix} \hat{D} & [0] \\ [0] & Q \end{bmatrix} \quad (8.64)$$

the covariance time update can be expressed as

$$\tilde{P} = W D W^T \quad (8.65)$$

The covariance time update is accomplished by applying Gram–Schmidt orthogonalization to determine the row vectors of the matrix  $W$ . The algorithm for the time update can be found in Bierman [3].

In many applications, the  $U$ - $D$  factor time update is not used. After its initialization and propagation to the measurement time using Eq. (8.61), the error covariance  $P$  matrix is factored for the measurement update. After completing the measurement update in factored form, the factors are recombined into the  $P$  matrix and again propagated to the next measurement time, whereupon the cycle is repeated.

### Example 8.3 U-D Measurement Update Algorithm

The  $U$ - $D$  measurement update algorithm is developed by illustration, using the notation as just defined, for the  $3 \times 3$  example problem presented in Example 2.1. The terms within the brackets in Eq. (8.58) become

$$\left[ \tilde{D} - \frac{1}{\alpha} \mathbf{g} \mathbf{g}^T \right] = \begin{bmatrix} d_1 - \frac{1}{\alpha} g_1 g_1 & -\frac{1}{\alpha} g_1 g_2 & -\frac{1}{\alpha} g_1 g_3 \\ -\frac{1}{\alpha} g_2 g_1 & d_2 - \frac{1}{\alpha} g_2 g_2 & -\frac{1}{\alpha} g_2 g_3 \\ -\frac{1}{\alpha} g_3 g_1 & -\frac{1}{\alpha} g_3 g_2 & d_3 - \frac{1}{\alpha} g_3 g_3 \end{bmatrix}$$

The product of this matrix with the  $U$  factors, as shown in Eq. (8.58), yields the updated factors. Equating the measurement update form of the  $U$ - $D$  factors with (8.58) yields

$$\hat{U} \hat{D} \hat{U}^T = \tilde{U} \left[ \tilde{D} - \frac{1}{\alpha} \mathbf{g} \mathbf{g}^T \right] \tilde{U}^T$$

Equating the corresponding elements in the bottom rows on the left- and right-hand sides in this equation yields the following equations:

$$\begin{aligned}\hat{d}_3 &= \tilde{d}_3 - \frac{1}{\alpha} g_3 g_3 \\ \hat{d}_3 \hat{u}_{23} &= -\frac{1}{\alpha} g_3 g_2 + \left( \tilde{d}_3 - \frac{1}{\alpha} g_3 g_3 \right) \tilde{u}_{23} \\ \hat{d}_3 \hat{u}_{13} &= -\frac{1}{\alpha} g_3 g_1 - \frac{1}{\alpha} g_3 g_2 \tilde{u}_{12} + \left( \tilde{d}_3 - \frac{1}{\alpha} g_3 g_3 \right) \tilde{u}_{13}\end{aligned}$$

The nomenclature defined earlier allows the expression for the measurement covariance as

$$\alpha = \mathbf{f}^T \mathbf{g} + r$$

or

$$\alpha_j = \sum_{i=1}^j f_i g_i + r$$

where the subscripted  $\alpha$  is an incremental sum and  $\alpha_0 = r$ .

Using the definitions in Eqs. (8.56) and (8.57), the first of the preceding equations becomes

$$\begin{aligned}\hat{d}_3 &= \tilde{d}_3 - \frac{1}{\alpha} g_3 g_3 & \left\{ g_3 = \tilde{d}_3 f_3 \right\} \\ &= \tilde{d}_3 - \frac{\tilde{d}_3}{\alpha} f_3 g_3 \\ &= \tilde{d}_3 \left( \frac{\alpha - f_3 g_3}{\alpha} \right) & \left\{ \alpha = \alpha_3 = r + f_1 g_1 + f_2 g_2 + f_3 g_3 \right\} \\ &= \tilde{d}_3 \frac{\alpha_2}{\alpha}\end{aligned}$$

Continuing with the other two equations and using the result just obtained:

$$\begin{aligned}\hat{d}_3 \hat{u}_{23} &= -\frac{1}{\alpha} g_3 g_2 + \left( \tilde{d}_3 - \frac{1}{\alpha} g_3 g_3 \right) \tilde{u}_{23} \\ &= -\frac{1}{\alpha} g_3 g_2 + \hat{d}_3 \tilde{u}_{23} & \left\{ \frac{\alpha}{\alpha_2} = \frac{\tilde{d}_3}{\hat{d}_3} \right\}\end{aligned}$$

or,

$$\hat{u}_{23} = \tilde{u}_{23} - \frac{f_3}{\alpha_2} g_2$$

And

$$\begin{aligned}\hat{d}_3\hat{u}_{13} &= -\frac{1}{\alpha}g_3g_1 - \frac{1}{\alpha}g_3g_2\tilde{u}_{12} + \left(\tilde{d}_3 - \frac{1}{\alpha}g_2g_3\right)\tilde{u}_{13} \\ &= -\frac{1}{\alpha}g_3(g_1 + g_2\tilde{u}_{12}) + \hat{d}_3\tilde{u}_{13}\end{aligned}$$

or

$$\hat{u}_{13} = \tilde{u}_{13} - \frac{f_3}{\alpha_2}(g_1 + g_2\tilde{u}_{12})$$

The 2:2 element of the resulting product in the preceding second equation is

$$\begin{aligned}\hat{d}_2 + \hat{u}_{23}\hat{d}_3\hat{u}_{23} &= \left(\tilde{d}_2 - \frac{1}{\alpha}g_2g_2\right) - \frac{1}{\alpha}g_2g_3\tilde{u}_{23} \\ &\quad - \frac{1}{\alpha}g_2g_3\tilde{u}_{23} + \left(\tilde{d}_3 - \frac{1}{\alpha}g_3g_3\right)\tilde{u}_{23}^2 \\ &= \left(\tilde{d}_2 - \frac{1}{\alpha}g_2g_2\right) - \frac{2}{\alpha}g_2g_3\tilde{u}_{23} + \tilde{u}_{23}\hat{d}_3\tilde{u}_{23}\end{aligned}$$

The second term on the left-hand side (LHS) of this equation, using the results obtained before, becomes

$$\hat{u}_{23}\hat{d}_3\hat{u}_{23} = \tilde{u}_{23}\hat{d}_3\tilde{u}_{23} - \frac{2}{\alpha}g_2g_3\tilde{u}_{23} + \frac{(g_3g_2/\alpha)(g_3g_2/\alpha)}{\hat{d}_3}$$

Cancelling applicable terms, the following equation is obtained:

$$\begin{aligned}\hat{d}_2 &= \left(\tilde{d}_2 - \frac{1}{\alpha}g_2g_2\right) - \frac{(g_2g_3/\alpha)(g_2g_3/\alpha)}{\hat{d}_3} & \{g_2 = \tilde{d}_2 f_2\} \\ &= \tilde{d}_2 \left( \frac{\alpha - f_2g_2}{\alpha} \right) - \frac{(\tilde{d}_2 f_2 g_2 / \alpha)(\tilde{d}_3 f_3 g_3 / \alpha)}{\hat{d}_3} \\ &= \tilde{d}_2 \left[ \alpha - f_2g_2 - \left( \frac{f_2g_2}{\alpha_2} \right) f_3 g_3 \right] \frac{1}{\alpha} & \{\alpha - f_2g_2 = \alpha_1 + f_3g_3\} \\ &= \tilde{d}_2 [(\alpha_1 + f_2g_2)(\alpha_1 + f_3g_3) - f_2g_2f_3g_3] \frac{1}{\alpha_2\alpha} \\ &= \tilde{d}_2 (\alpha_1 + f_2g_2 + f_3g_3) \frac{\alpha_1}{\alpha_2\alpha} & \{\alpha = \alpha_1 + f_2g_2 + f_3g_3\} \\ &= \tilde{d}_2 \frac{\alpha_1}{\alpha_2}\end{aligned}$$

By induction,

$$\hat{d}_j = \tilde{d}_j \frac{\alpha_{j-1}}{\alpha_j}$$

Finally, the 1:2 element of the resulting product in the preceding second equation is

$$\begin{aligned}\hat{d}_2\hat{u}_{12} + \hat{u}_{23}\hat{d}_3\hat{u}_{13} &= -\frac{1}{\alpha}g_2g_1 + \left(\tilde{d}_2 - \frac{1}{\alpha}g_2g_2\right)\tilde{u}_{12} - \frac{1}{\alpha}g_2g_3\tilde{u}_{13} \\ &\quad + \left[-\frac{1}{\alpha}g_3g_1 - \frac{1}{\alpha}g_3g_2\tilde{u}_{12} + \left(\tilde{d}_3 - \frac{1}{\alpha}g_3g_3\right)\tilde{u}_{13}\right]\tilde{u}_{23}\end{aligned}$$

The second term on the LHS is

$$\begin{aligned}\hat{u}_{23}\hat{d}_3\hat{u}_{13} &= \tilde{u}_{23}\hat{d}_3\tilde{u}_{13} - \frac{1}{\alpha}g_3g_2\tilde{u}_{13} - \frac{1}{\alpha}g_3g_1\tilde{u}_{23} \\ &\quad - \frac{1}{\alpha}g_3g_2\tilde{u}_{12}\tilde{u}_{23} + \frac{1}{\alpha^2}g_3^2g_2 \frac{(g_1 + g_2\tilde{u}_{12})}{\hat{d}_3}\end{aligned}$$

Cancelling applicable terms, the following equation is obtained:

$$\begin{aligned}\hat{d}_2\hat{u}_{12} &= -\frac{1}{\alpha}g_2g_1 + \left(\tilde{d}_2 - \frac{1}{\alpha}g_2g_2\right)\tilde{u}_{12} - \frac{1}{\alpha^2}g_3^2g_2 \frac{(g_1 + g_2\tilde{u}_{13})}{\hat{d}_3} \\ &= \hat{d}_2\tilde{u}_{12} - \frac{1}{\alpha}g_2g_1 - \frac{1}{\alpha^2} \frac{g_3^2g_2g_1}{\hat{d}_3} \\ &= \hat{d}_2\tilde{u}_{12} - \frac{1}{\alpha}g_2g_1 \left(1 + \frac{1}{\alpha} \frac{g_3^2}{\hat{d}_3}\right) && \left\{g_1 = \tilde{d}_1f_1\right\} \\ &= \hat{d}_2\tilde{u}_{12} - \frac{\tilde{d}_2}{\alpha} \tilde{d}_1f_1f_2 \frac{\left[\hat{d}_3 + (1/\alpha)g_3^2\right]}{\hat{d}_3} \\ &= \hat{d}_2\tilde{u}_{12} - \frac{\tilde{d}_2}{\alpha} \tilde{d}_1f_1f_2 \frac{\tilde{d}_3}{\hat{d}_3} \\ &= \hat{d}_2\tilde{u}_{12} - \frac{\tilde{d}_2}{\alpha_2} \tilde{d}_1f_1f_2 && \left\{\frac{\alpha_2}{\alpha_1} = \frac{\tilde{d}_2}{\hat{d}_2}\right\} \\ &= \hat{d}_2 \left( \tilde{u}_{12} - \frac{\tilde{d}_2}{\alpha_1} f_1f_2 \right)\end{aligned}$$

or,

$$\hat{u}_{12} = \tilde{u}_{12} - \frac{f_2}{\alpha_1}g_1$$

The results in this example are summarized here:

$$\begin{aligned}\hat{d}_1 &= \tilde{d}_1 \frac{\alpha_0}{\alpha_1} \\ \hat{d}_2 &= \tilde{d}_2 \frac{\alpha_1}{\alpha_2} \\ \hat{u}_{12} &= \tilde{u}_{12} - \frac{f_2}{\alpha_1} g_1 \\ \hat{d}_3 &= \tilde{d}_3 \frac{\alpha_2}{\alpha_3} \\ \hat{u}_{13} &= \tilde{u}_{13} - \frac{f_3}{\alpha_2} (g_1 + g_2 \tilde{u}_{12}) \\ \hat{u}_{23} &= \tilde{u}_{23} - \frac{f_3}{\alpha_2} g_2\end{aligned}$$

## 8.5 Summed Measurements

It was illustrated in Problem 2.1 that data averaging can improve the estimate of a constant quantity by reducing the estimation error covariance. This serves as motivation to consider using this method to reduce errors in noisy measurements; thus, this concept is extended in this section to allow for time-varying states. This concept is also useful in changing a measurement's state relationship, for example, using change in position over a time interval as measurements, which is then related to an accumulation of velocity error states over that same interval. Both of these concepts can be formulated into a common structure of summed measurements. Developments that follow address the use of summed measurements over a given time interval prior to their being processed in the Kalman filter measurement update equation. It is shown that the use of summed measurements requires modification to the algorithm's measurement matrix and measurement covariances.

The time dependence of measurements is indicated next by using the subscript  $k$ :

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{v}_k$$

Summing  $m$  of these measurements in between measurement update cycles yields

$$\sum_{k=1}^m \mathbf{z}_k \Delta = \sum_{k=1}^m H_k \mathbf{x}_k \Delta + \sum_{k=1}^m \mathbf{v}_k \Delta \quad (8.66)$$

where a temporary time-independent scalar quantity  $\Delta$  has been multiplied to each of the terms in the measurement equation. The definition of this scalar

quantity will be specified later depending on which of the preceding two cases is to be illustrated.

It is assumed that the measurement matrix  $H_k$  does not have significant time dependence during this summation process. The first term on the RHS of Eq. (8.66) is rewritten as

$$\begin{aligned}\sum H_k \mathbf{x}_k \Delta &= H\Delta \sum \mathbf{x}_k \\ &= H\Delta[\mathbf{x}(t_1) + \mathbf{x}(t_2) + \cdots + \mathbf{x}(t_m)] \\ &= H\Delta[\Phi(t_1, t_0)\mathbf{x}(t_0) + \Phi(t_2, t_0)\mathbf{x}(t_0) + \cdots + \Phi(t_m, t_0)\mathbf{x}(t_0)] \\ &= H\Delta[\Phi(t_1, t_0) + \Phi(t_2, t_0) + \cdots + \Phi(t_m, t_0)]\mathbf{x}(t_0)\end{aligned}\quad (8.67)$$

where  $\mathbf{x}(t_0)$  is the state vector prior to beginning of the summation interval. The system model dynamics are assumed to be time invariant, and the time intervals are equal length such that the preceding state transition matrices can be expressed as

$$\begin{aligned}\Phi(t_1, t_0) &= e^{F\Delta t}; \quad \Delta t = t_1 - t_0 \\ \Phi(t_2, t_0) &= e^{2F\Delta t}; \quad 2\Delta t = t_2 - t_0 \\ \Phi(t_m, t_0) &= e^{mF\Delta t}; \quad m\Delta t = t_m - t_0\end{aligned}$$

Substituting these expressions into the preceding equation yields

$$\sum H_k \mathbf{x}_k \Delta = H\Delta[e^{F\Delta t} + e^{2F\Delta t} + \cdots + e^{mF\Delta t}]\mathbf{x}(t_0)$$

The terms in the brackets in this equation represent a geometric series whose ratio of the current term to the preceding term is  $e^{F\Delta t}$ . Rewriting this equation using the series summation results in

$$\sum H_k \mathbf{x}_k \Delta = H\Delta F^{-1}[e^{mF\Delta t} - I] \frac{1}{\Delta t} \mathbf{x}(t_0) \quad (8.68)$$

With equal time intervals, the total time interval, referenced to the start time, used for the measurement summation can be defined as  $\Delta T = m\Delta t$ . Assuming that the exponential in Eq. (8.68) can be approximated by a truncated series with three

terms, this equation is rewritten as

$$\begin{aligned}
 \sum H_k \mathbf{x}_k \Delta &= H \Delta F^{-1} \left[ \left( I + F \Delta T + F^2 \frac{\Delta T}{2!} + \dots \right) - I \right] \frac{1}{\Delta t} \mathbf{x}(t_0) \\
 &\approx H \Delta F^{-1} \left[ F \Delta T + F^2 \frac{\Delta T^2}{2!} \right] \frac{1}{\Delta t} \mathbf{x}(t_0) \\
 &\approx H \Delta \left[ \Delta T + F \frac{\Delta T^2}{2!} \right] \frac{1}{\Delta t} \mathbf{x}(t_0) \\
 &\approx H \Delta \left[ I + F \frac{\Delta T}{2!} \right] \frac{\Delta T}{\Delta t} \mathbf{x}(t_0) \\
 &\approx H \Delta m \left[ I + F \frac{\Delta T}{2!} \right] \mathbf{x}(t_0)
 \end{aligned} \tag{8.69}$$

### 8.5.1 Data Averaging

Returning to Eq. (8.66), using the results in Eq. (8.69) and defining  $\Delta$  as

$$\Delta = \frac{1}{m}$$

and redefining the start time as the current time, yields

$$\frac{1}{m} \sum \mathbf{z}_k = H \left[ I - F \frac{\Delta T}{2!} \right] \mathbf{x}(t) + \frac{1}{m} \sum \mathbf{v}_k$$

Or defining the summations as a new measurement and equivalent noise,

$$\xi_k = H \left[ I - F \frac{\Delta T}{2!} \right] \mathbf{x}(t) + \mathbf{v}_k \tag{8.70}$$

where

$$\xi_k = \frac{1}{m} \sum \mathbf{z}_k \tag{8.71}$$

and

$$\mathbf{v}_k = \frac{1}{m} \sum \mathbf{v}_k \tag{8.72}$$

The covariance of the measurement noise  $\nu$  is adjusted by

$$R_\nu = \frac{1}{m} R_\nu \quad (8.73)$$

### 8.5.2 Numerical Integration

Again returning to Eq. (8.66), using the results in Eq. (8.69) and now defining  $\Delta$  as

$$\Delta = \Delta t$$

yield

$$\sum z_k \Delta t = H \Delta t m \left[ I - F \frac{\Delta T}{2!} \right] x(t) + \sum v_k \Delta t$$

Or defining the summations as a new measurement and equivalent noise,

$$\xi_k = H \Delta T \left[ I - F \frac{\Delta T}{2!} \right] x(t) + v_k \quad (8.74)$$

where

$$\xi_k = \Delta t \sum z_k \quad (8.75)$$

and

$$v_k = \Delta t \sum v_k \quad (8.76)$$

The covariance of the measurement noise  $\nu$  is adjusted by

$$R_\nu = \frac{\Delta T^2}{m} R_\nu \quad (8.77)$$

## 8.6 Combined Estimate from Two (or More) Kalman Filters

If two or more Kalman filters are providing estimates of related navigation states, not necessarily using the same state vector structure, these estimates can be combined to establish an unbiased estimate. An example of the application of this concept is when there are two navigation systems, for example, either two inertial systems or one inertial and one dead-reckoning system located on a vehicle and separate Kalman filter algorithms are implemented for each system. If the two sets of navigation states are not the same but can be related by transformation, then the concept developed in this section can be applied. This concept can also be extended for combining estimates from more than two Kalman filters.

Define an estimate from two Kalman filters as the following weighted combination:

$$\hat{\mathbf{x}}_{1/2} \equiv W_1 \hat{\mathbf{x}}_1 + W_2 \hat{\mathbf{x}}_2 \quad (8.78)$$

where the estimate from the  $i$ th filter is related to the true state value by

$$\hat{\mathbf{x}}_i = \mathbf{x} + \varepsilon \mathbf{x}_i \quad (8.79)$$

and where each filter's estimate is assumed to be unbiased— $E[\varepsilon \mathbf{x}_i] = 0$ . The relationship between the weights can be determined by requiring the combined estimation error to be unbiased—zero expected value. This combined estimation error is written as

$$\varepsilon \mathbf{x}_{1/2} = \hat{\mathbf{x}}_{1/2} - \mathbf{x} \quad (8.80)$$

and the expectation of this error becomes

$$\begin{aligned} E[\varepsilon \mathbf{x}_{1/2}] &= E[W_1(\mathbf{x} + \varepsilon \mathbf{x}_1) + W_2(\mathbf{x} + \varepsilon \mathbf{x}_2) - \mathbf{x}] \\ &= W_1 E[\mathbf{x}] + W_2 E[\mathbf{x}] - E[\mathbf{x}] \end{aligned}$$

which implies

$$W_2 = I - W_1 \quad \forall \quad E[\mathbf{x}] \neq 0 \quad (8.81)$$

for an unbiased estimate.

The specific relationship for the preceding weights is determined by minimizing the covariance, thus establishing a minimum variance estimate. The error covariance of the combined estimate is expressed as

$$\begin{aligned} P_{1/2} &= E\left[\varepsilon \mathbf{x}_{1/2} \varepsilon \mathbf{x}_{1/2}^T\right] \\ &= E\left\{[W_1(\mathbf{x} + \varepsilon \mathbf{x}_1) + W_2(\mathbf{x} + \varepsilon \mathbf{x}_2) - \mathbf{x}][W_1(\mathbf{x} + \varepsilon \mathbf{x}_1) \right. \\ &\quad \left. + W_2(\mathbf{x} + \varepsilon \mathbf{x}_2) - \mathbf{x}]^T\right\} \\ &= W_1 E[\varepsilon \mathbf{x}_1 \varepsilon \mathbf{x}_1^T] W_1^T + W_2 E[\varepsilon \mathbf{x}_2 \varepsilon \mathbf{x}_2^T] W_2^T \\ &\quad + W_2 E[\varepsilon \mathbf{x}_2 \varepsilon \mathbf{x}_1^T] W_1^T + W_1 E[\varepsilon \mathbf{x}_1 \varepsilon \mathbf{x}_2^T] W_2^T + O[E(\varepsilon \mathbf{x}_i)] \end{aligned}$$

Assuming that individual filter errors are uncorrelated,

$$E\left[\varepsilon \mathbf{x}_i \varepsilon \mathbf{x}_j^T\right] = [0] \quad \forall \quad i \neq j \quad (8.82)$$

Then the combined covariance becomes

$$\begin{aligned} P_{1/2} &= W_1 P_1 W_1^T + W_2 P_2 W_2^T \\ &= W_1 P_1 W_1^T + (I - W_1) P_2 (I - W_1)^T \end{aligned} \quad (8.83)$$

Weighting matrices are determined by taking the variation of this equation and setting the result to the null matrix. The variation with respect to the weighting matrix becomes

$$\begin{aligned} \delta P_{1/2} &= \delta [W_1 P_1 W_1^T + P_2 - W_1 P_2 - P_2 W_1^T + W_1 P_2 W_1^T] \\ &= \delta W_1 P_1 W_1^T + W_1 P_1 \delta W_1^T - \delta W_1 P_2 - P_2 \delta W_1^T \\ &\quad + \delta W_1 P_2 W_1^T + W_1 P_2 \delta W_1^T \\ &= \delta W_1 [P_1 W_1^T - P_2 + P_2 W_1^T] + [W_1 P_1 - P_2 + W_1 P_2] \delta W_1^T \end{aligned}$$

setting this result to the null yields, for any  $\delta W_1 \neq [0]$

$$[P_1 W_1^T - P_2 + P_2 W_1^T] = [0] = [W_1 P_1 - P_2 + W_1 P_2]$$

or

$$W_1 = P_2(P_1 + P_2)^{-1} \quad (8.84)$$

and, using the relationship between the preceding weighing matrices

$$W_2 = P_1(P_1 + P_2)^{-1} \quad (8.85)$$

Substituting Eqs. (8.84) and (8.85) into the combined covariance matrix equation (8.83) yields

$$\begin{aligned} P_{1/2} &= W_1 P_1 W_1^T + P_2 - W_1 P_2 - P_2 W_1^T + W_1 P_2 W_1^T \\ &= P_2(P_1 + P_2)^{-1} P_1 (P_1 + P_2)^{-1} P_2 + P_2 - P_2 (P_1 + P_2)^{-1} P_2 \\ &\quad - P_2 (P_1 + P_2)^{-1} P_2 + P_2 (P_1 + P_2)^{-1} P_2 (P_1 + P_2)^{-1} P_2 \\ &= P_2 (P_1 + P_2)^{-1} (P_1 + P_2) (P_1 + P_2)^{-1} P_2 \\ &\quad + P_2 - P_2 (P_1 + P_2)^{-1} P_2 - P_2 (P_1 + P_2)^{-1} P_2 \\ &= P_2 (P_1 + P_2)^{-1} P_2 + P_2 - 2 P_2 (P_1 + P_2)^{-1} P_2 \\ &= P_2 [(P_1 + P_2)^{-1} + P_2^{-1} - 2(P_1 + P_2)^{-1}] P_2 \\ &= P_2 [P_2^{-1} - (P_1 + P_2)^{-1}] P_2 \\ &= P_2 - P_2 (P_1 + P_2)^{-1} P_2 \end{aligned}$$

or, using the matrix inversion lemma from Chapter 2 (see Example 2.2), the combined estimation error covariance becomes

$$P_{1/2}^{-1} = P_1^{-1} + P_2^{-1} \quad (8.86)$$

The combined estimate in Eq. (8.78) becomes

$$\hat{\mathbf{x}}_{1/2} = P_2(P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + P_1(P_1 + P_2)^{-1}\hat{\mathbf{x}}_2 \quad (8.87)$$

This equation can be rewritten in a more compact form. Premultiplying this equation with Eq. (8.86) yields

$$\begin{aligned} P_{1/2}^{-1}\hat{\mathbf{x}}_{1/2} &= (P_1^{-1} + P_2^{-1})[P_2(P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + P_1(P_1 + P_2)^{-1}\hat{\mathbf{x}}_2] \\ &= P_1^{-1}[P_2(P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + P_1(P_1 + P_2)^{-1}\hat{\mathbf{x}}_2] \\ &\quad + P_2^{-1}[P_2(P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + P_1(P_1 + P_2)^{-1}\hat{\mathbf{x}}_2] \\ &= P_1^{-1}P_2(P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + (P_1 + P_2)^{-1}\hat{\mathbf{x}}_2 \\ &\quad + (P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + P_2^{-1}P_1(P_1 + P_2)^{-1}\hat{\mathbf{x}}_2 \\ &= (P_1^{-1}P_2 + I)(P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + (P_2^{-1}P_1 + I)(P_1 + P_2)^{-1}\hat{\mathbf{x}}_2 \\ &= P_1^{-1}(P_2 + P_1)(P_1 + P_2)^{-1}\hat{\mathbf{x}}_1 + P_2^{-1}(P_1 + P_2)(P_1 + P_2)^{-1}\hat{\mathbf{x}}_2 \end{aligned}$$

Or,

$$P_{1/2}^{-1}\hat{\mathbf{x}}_{1/2} = P_1^{-1}\hat{\mathbf{x}}_1 + P_2^{-1}\hat{\mathbf{x}}_2 \quad (8.88)$$

Extending this concept to more than two filters can be easily followed from the form of Eqs. (8.86) and (8.88).

## 8.7 Derivative-Free Estimation

### 8.7.1 Unscented Kalman Filter

Table 8.2 summarizes the unscented Kalman filter algorithm [11–13]. The algorithm begins with forming additional vectors about the state vector that are displaced, or dispersed, using column vectors form by the error covariance matrix  $P$  factor (see Problem 2.15). In this table, the symbol  $\sqrt{i}$  represents the scaled  $i$  th column/row vector of the covariance matrix factor. Once established, these new vectors are instantiated using a discrete nonlinear form of the system dynamics model. From this result, a mean of the vectors is computed based on a weighted sum, and a covariance about this mean is computed, again using a weighted sum.

Each of the vectors instantiated via the system model is instantiated using a nonlinear form of the measurement model. In similar fashion with the dynamics instantiation, the results from the measurement model instantiation,

**Table 8.2 Unscented/Kalman filter [11–14]<sup>a</sup>**

Establish symmetric sigma points about state estimate	$\hat{\chi}_0 \equiv \hat{x}$ $\hat{\chi}_i = \hat{x} + \sqrt{(n + \kappa)P_i} \quad \forall i = 1, \dots, n$ $\hat{\chi}_i = \hat{x} - \sqrt{(n + \kappa)P_i} \quad \forall i = n + 1, \dots, 2n$	$W_0 = \kappa/n + \kappa$ and $W_i = \frac{1}{2(n + \kappa)}$ $\kappa = \text{constant}$ , $n = \text{number of states}$ $\sqrt{(n + \kappa)P_i} = \text{scaled } i\text{th } P \text{ matrix row/column of square-root factor}$
Instantiate sigma points through process model	$\tilde{\chi}_i = f[\hat{\chi}_i]$	Discrete form of nonlinear dynamics equation for time propagation
Predict mean	$\bar{x} = \sum_{i=0}^{2n} W_i \tilde{\chi}_i$	Weighted linear combination
Predict covariance	$\bar{P}_{xx} = \sum_{i=0}^{2n} W_i [\tilde{\chi}_i - \bar{x}] \times [\tilde{\chi}_i - \bar{x}]^T \{+Q\}$	Weighted linear combination of deviations from predicted mean
Instantiate sigma points through measurement model	$Y_i = h[\tilde{\chi}_i]$	Nonlinear measurement model
Predict observation	$\bar{y} = \sum_{i=0}^{2n} W_i Y_i$	Weighted linear combination
Innovation covariance	$P_{yy} = \sum_{i=0}^{2n} W_i (Y_i - \bar{y})(Y_i - \bar{y})^T \{+R\}$	Weighted linear combination of deviations from mean observation
Cross covariance	$P_{xy} = \sum_{i=0}^{2n} W_i (\tilde{\chi}_i - \bar{x})(Y_i - \bar{y})^T$	Weighted linear combination of deviations
Kalman filter equations	$K = P_{xy} P_{yy}^{-1}$ $\hat{x} = \bar{x} + K(y - \bar{y})$ $P = \bar{P}_{xx} - K P_{yy} K^T$	—

<sup>a</sup>This form of the UKF algorithm reflects the weights defined in the earliest of the references cited.

for each of the dispersed state vectors, are used to compute a mean and covariance. A cross covariance is also computed. The weighted means and covariances thus computed are used in a form of the Kalman filter equations (see Problem 8.9). The process noise and measurement covariance matrices are inserted at the appropriate computational stage in the table.

The authors of this algorithm have demonstrated that the covariance matrix formed with the weighted sum, using the weights presented in Table 8.2, reproduces the error covariance matrix. This is demonstrated in the sequence of steps in the following equation. Note, the caret or tilde symbol has been temporarily ignored.

$$\begin{aligned}
 P &= W_0(\chi_0 - x)(\chi_0 - x)^T \\
 &\quad + W_1(\chi_1 - x)(\chi_1 - x)^T + W_2(\chi_2 - x)(\chi_2 - x)^T + \dots \\
 &\quad + W_n(\chi_n - x)(\chi_n - x)^T \\
 &\quad + W_{n+1}(\chi_{n+1} - x)(\chi_{n+1} - x)^T + W_{n+2}(\chi_{n+2} - x)(\chi_{n+2} - x)^T + \dots \\
 &\quad + W_{2n}(\chi_{2n} - x)(\chi_{2n} - x)^T \\
 &= 0 \\
 &\quad + W_1(x + \sqrt{\chi}_1 - x)(x + \sqrt{\chi}_1 - x)^T \\
 &\quad + W_2(x + \sqrt{\chi}_2 - x)(x + \sqrt{\chi}_2 - x)^T + \dots \\
 &\quad + W_n(x + \sqrt{\chi}_n - x)(x + \sqrt{\chi}_n - x)^T \\
 &\quad + W_{n+1}(x - \sqrt{\chi}_{n+1} - x)(x - \sqrt{\chi}_{n+1} - x)^T \\
 &\quad + W_{n+2}(x - \sqrt{\chi}_{n+2} - x)(x - \sqrt{\chi}_{n+2} - x)^T + \dots \\
 &\quad + W_{2n}(x - \sqrt{\chi}_{2n} - x)(x - \sqrt{\chi}_{2n} - x)^T \\
 &= W_1(\sqrt{\chi}_1)^T(\sqrt{\chi}_1)^T + W_2(\sqrt{\chi}_2)^T(\sqrt{\chi}_2)^T + \dots + W_n(\sqrt{\chi}_n)^T(\sqrt{\chi}_n)^T \\
 &\quad + W_{n+1}(\sqrt{\chi}_{n+1})^T(\sqrt{\chi}_{n+1})^T + W_{n+2}(\sqrt{\chi}_{n+2})^T(\sqrt{\chi}_{n+2})^T \\
 &\quad + \dots + W_{2n}(\sqrt{\chi}_{2n})^T(\sqrt{\chi}_{2n})^T \\
 &= 2W_i \sum_{i=1}^n \left[ \sqrt{(n+\kappa)P_i} \right] \left[ \sqrt{(n+\kappa)P_i} \right]^T \\
 &= 2W_i(n+\kappa) \sum_{i=1}^n (\sqrt{P_i})(\sqrt{P_i})^T \\
 &= P
 \end{aligned} \tag{8.89}$$

where the symbol  $\sqrt{\cdot}_i$  represents the scaled  $i$ th column/row vector of the covariance matrix factor.

The following example presents a motivation for the UKF to illustrate the limitations of a linear Kalman filter for nonlinear systems. Using a scalar nonlinear function, a variance is formed that leads to an infinite series that is used as “truth,” and matching the order of its terms relative to an approximation to this variance formed using differences rather than derivatives.

### Example 8.4 Scalar Example for True Covariance Approximation

This example illustrates the formulation of the covariance of a nonlinear function by expanding this function in a Taylor series. The inputs to this function consist of a variable’s mean plus its random variation, and the expected value of this variation odd powers is assumed to be zero. Consider the nonlinear function

$$y = f(x)$$

The input to this function is the mean value plus random variable deviation.

$$x = \bar{x} + \delta x$$

where it is assumed that the random variable deviation satisfies the following:

$$E[\delta x] \equiv 0$$

$$E[\delta x^2] \equiv P_{xx}$$

and all odd powers of higher moments are zero

$$E[\delta x^3] = E[\delta x^5] = \dots \equiv 0$$

Expanding in Taylor series about the mean value yields

$$\begin{aligned} y &= f(\bar{x} + \delta x) \\ &= f(\bar{x}) + \frac{\partial f}{\partial x} \delta x + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} \delta x^3 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots \end{aligned}$$

Taking the expected value of this function, using the assumption concerning the random variable deviation’s odd powers, to form a mean of the function yields

$$\begin{aligned} \bar{y} &= E[f(\bar{x} + \delta x)] \\ &= E[f(\bar{x})] + E\left[\frac{\partial f}{\partial x} \delta x + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} \delta x^3 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots\right] \\ &= f(\bar{x}) + E\left[\frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots\right] \end{aligned}$$

The difference between function and expected value becomes

$$\begin{aligned}
 y - \bar{y} &= f(\bar{x} + \delta x) - E[f(\bar{x} + \delta x)] \\
 &= f(\bar{x}) + \frac{\partial f}{\partial x} \delta x + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} \delta x^3 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots \\
 &\quad - f(\bar{x}) - E\left[\frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots\right] \\
 &= \frac{\partial f}{\partial x} \delta x + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} \delta x^3 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots \\
 &\quad - E\left[\frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots\right]
 \end{aligned}$$

Then the variance of this difference is formed as

$$\begin{aligned}
 E[(y - \bar{y})^2] &= E[\{f(\bar{x} + \delta x) - E[f(\bar{x} + \delta x)]\}^2] \\
 &= E\left[\left(\frac{\partial f}{\partial x} \delta x + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} \delta x^3 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots\right.\right. \\
 &\quad \left.\left.- E\left[\frac{1}{2!} \frac{\partial^2 f}{\partial x^2} \delta x^2 + \frac{1}{4!} \frac{\partial^4 f}{\partial x^4} \delta x^4 + \dots\right]\right)^2\right] \\
 &= E\left[\left.f_x^2 \delta x^2 + \frac{1}{2!} f_{xx} f_x \delta x^3 + \frac{1}{3!} f_{xxx} f_x \delta x^4\right.\right. \\
 &\quad \left.\left.+ \dots - \frac{1}{2!} f_{xx} f_x E[\delta x^2] \delta x - \dots\right.\right. \\
 &\quad \left.\left.+ \frac{1}{2!} f_x f_{xx} \delta x^3 + \frac{1}{2!2!} f_{xx}^2 \delta x^4\right.\right. \\
 &\quad \left.\left.+ \dots - \frac{1}{2!2!} f_{xx}^2 E[\delta x^2] \delta x^2 - \dots\right.\right. \\
 &\quad \left.\left.+ \frac{1}{3!} f_x f_{xxx} \delta x^4 + \dots - \dots\right.\right. \\
 &\quad \left.\left.- \frac{1}{2!} f_{xx} f_x E[\delta x^2] \delta x - \dots\right.\right. \\
 &\quad \left.\left.- \frac{1}{2!2!} f_{xx}^2 E[\delta x^2] \delta x^2 - \dots\right.\right] \\
 &= f_x^2 E[\delta x^2] + \frac{1}{3!} f_{xxx} f_x E[\delta x^4] + \frac{1}{2!2!} f_{xx}^2 E[\delta x^4] + \frac{1}{3!} f_x f_{xxx} E[\delta x^4] \\
 &\quad - \frac{1}{2!2!} f_{xx}^2 E[\delta x^2] E[\delta x^2] + \dots
 \end{aligned}$$

where the notation  $f_x$  implies the operation  $\partial f / \partial x$  and so forth for  $f_{xx}$ , etc.. The variance of the nonlinear function based on the distribution of the input's random variable deviation becomes

$$\begin{aligned} P_{yy} = & f_x^2 P_{xx} + E\left[\frac{1}{3!} f_{xxx} f_x \delta x^4 + \frac{1}{2 * 2!} f_{xx}^2 \delta x^4 + \frac{1}{3!} f_x f_{xxx} \delta x^4\right] \\ & - E\left[\frac{1}{2!} f_{xx} \delta x^2\right] E\left[\frac{1}{2!} f_{xx} \delta x^2\right] + \dots \end{aligned}$$

Linear system function would include only the first term. For a nonlinear system function, this expression for the true variance is finite only when the derivatives it contains become zero.

Rationale for the formulation of the UKF algorithm is presented by way of another example. This example also considers a scalar nonlinear system and presents an expansion for the predicted variance listed in Table 8.2.

### Example 8.5 Scalar Nonlinear System Example

Consider the scalar system and measurement models

$$x_{i+1} = f(x_i) + w_i \quad \text{and} \quad z_j = h(x_j) + v_j$$

In this example, the number of states is one ( $n = 1$ ), and the weights in Table 8.2 become

$$W_0 = \frac{\kappa}{\kappa + 1}, \quad W_1 = \frac{1}{2(\kappa + 1)}, \quad \text{and} \quad W_2 = \frac{1}{2(\kappa + 1)}$$

The first step in the sequence of steps is to form the dispersed states

$$\hat{x}_0 \equiv \hat{x} \equiv x_0, \quad \hat{x}_1 = \hat{x} + \sqrt{\kappa + 1}\sigma, \quad \text{and} \quad \hat{x}_2 = \hat{x} - \sqrt{\kappa + 1}\sigma$$

where the state  $\hat{x}$  and  $\sigma$  are both dynamic variables. In the following, the root in the preceding terms will be represented symbolically as simply  $\sqrt{\cdot}$ .

The predicted mean is formed by instantiating the dispersed state as

$$\begin{aligned} \bar{x} &= W_0 \tilde{x}_0 + W_1 \tilde{x}_1 + W_2 \tilde{x}_2 \\ &= \frac{\kappa}{\kappa + 1} f(\hat{x}_0) + \frac{1}{2(\kappa + 1)} f(\hat{x}_1) + \frac{1}{2(\kappa + 1)} f(\hat{x}_2) \\ &= \frac{1}{\kappa + 1} \{ \kappa f(\hat{x}_0) + \frac{1}{2} [f(\hat{x} + \sqrt{\sigma}) + f(\hat{x} - \sqrt{\sigma})] \} \end{aligned}$$

Continuing with the variance about the predicted mean

$$\begin{aligned} \bar{P}_{xx} &= \sum_{i=0}^2 W_i [\tilde{x}_i - \bar{x}] [\tilde{x}_i - \bar{x}]^T \\ &= W_0 (\tilde{x}_0 - \bar{x})^2 + W_1 (\tilde{x}_1 - \bar{x})^2 + W_2 (\tilde{x}_2 - \bar{x})^2 \end{aligned}$$

$$\begin{aligned}
&= \frac{\kappa}{\kappa+1} \left( f(\hat{x}) - \frac{1}{\kappa+1} \left\{ \kappa f(\hat{x}) + \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) + f(\hat{x} - \sqrt{\sigma})] \right\} \right)^2 \\
&\quad + \frac{1}{2(\kappa+1)} \left( f(\hat{x} + \sqrt{\sigma}) - \frac{1}{\kappa+1} \left\{ \kappa f(\hat{x}) + \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) + f(\hat{x} - \sqrt{\sigma})] \right\} \right)^2 \\
&\quad + \frac{1}{2(\kappa+1)} \left( f(\hat{x} - \sqrt{\sigma}) - \frac{1}{\kappa+1} \left\{ \kappa f(\hat{x}) + \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) + f(\hat{x} - \sqrt{\sigma})] \right\} \right)^2 \\
&= \frac{\kappa}{(\kappa+1)^2} \left\{ (\kappa+1)f(\hat{x}) - \kappa f(\hat{x}) - \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) + f(\hat{x} - \sqrt{\sigma})] \right\}^2 \\
&\quad + \frac{1}{2(\kappa+1)^2} \left\{ (\kappa+1)f(\hat{x} + \sqrt{\sigma}) - \kappa f(\hat{x}) - \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) + f(\hat{x} - \sqrt{\sigma})] \right\}^2 \\
&\quad + \frac{1}{2(\kappa+1)^2} \left\{ (\kappa+1)f(\hat{x} - \sqrt{\sigma}) - \kappa f(\hat{x}) - \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) + f(\hat{x} - \sqrt{\sigma})] \right\}^2
\end{aligned}$$

This expression can be simplified further to yield

$$\begin{aligned}
\bar{P}_{xx} &= \frac{\kappa}{2(\kappa+1)^2} \left\{ f(\hat{x} + \sqrt{\sigma}) - 2f(\hat{x}) + f(\hat{x} - \sqrt{\sigma}) \right\}^2 \\
&\quad + \frac{1}{2(\kappa+1)^2} \left\{ \kappa[f(\hat{x} + \sqrt{\sigma}) - f(\hat{x})] + \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) - f(\hat{x} - \sqrt{\sigma})] \right\}^2 \\
&\quad + \frac{1}{2(\kappa+1)^2} \left\{ \kappa[f(\hat{x}) - f(\hat{x} - \sqrt{\sigma})] + \frac{1}{2}[f(\hat{x} + \sqrt{\sigma}) - f(\hat{x} - \sqrt{\sigma})] \right\}^2
\end{aligned}$$

Examining the last terms in the second of the {} brackets, consider a difference between the following Taylor-series expansions about the state  $\hat{x}$ :

$$f(x + \sqrt{\sigma}) = f(x) + \frac{\partial f}{\partial x} \sqrt{\sigma} + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} (\sqrt{\sigma})^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} (\sqrt{\sigma})^3 + \dots$$

and

$$f(x - \sqrt{\sigma}) = f(x) - \frac{\partial f}{\partial x} \sqrt{\sigma} + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} (\sqrt{\sigma})^2 - \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} (\sqrt{\sigma})^3 + \dots$$

therefore

$$f(x + \sqrt{\sigma}) - f(x - \sqrt{\sigma}) = 2 \frac{\partial f}{\partial x} \sqrt{\sigma} + 2 \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} (\sqrt{\sigma})^3 + \dots$$

Or

$$\frac{1}{2}[f(x + \sqrt{\sigma}) - f(x - \sqrt{\sigma})] \approx \frac{\partial f}{\partial x} \sqrt{\sigma} + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} (\sqrt{\sigma})^3$$

Also, from these expansions, the term in the first { } bracket can be expressed as

$$[f(x + \sqrt{\sigma}) - 2f(x) + f(x - \sqrt{\sigma})] \approx 2 \frac{(\sqrt{\sigma})^2}{2!} \frac{\partial^2 f}{\partial x^2}$$

and the first terms in the second { } brackets become

$$f(x + \sqrt{\sigma}) - f(x) = \frac{\partial f}{\partial x} \sqrt{\sigma} + \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} (\sqrt{\sigma})^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} (\sqrt{\sigma})^3 + \dots$$

and

$$f(x) - f(x - \sqrt{\sigma}) = \frac{\partial f}{\partial x} \sqrt{\sigma} - \frac{1}{2!} \frac{\partial^2 f}{\partial x^2} (\sqrt{\sigma})^2 + \frac{1}{3!} \frac{\partial^3 f}{\partial x^3} (\sqrt{\sigma})^3 - \dots$$

Substituting these approximating expansions in the preceding variance equation yields

$$\begin{aligned} \bar{P}_{xx} &\approx \frac{\kappa}{2(\kappa+1)^2} \left\{ 2 \frac{(\sqrt{\sigma})^2}{2!} f_{xx} \right\}^2 \\ &+ \frac{1}{2(\kappa+1)^2} \left\{ \kappa \left[ (\sqrt{\sigma}) f_x + \frac{(\sqrt{\sigma})^2}{2!} f_{xx} + \frac{(\sqrt{\sigma})^3}{3!} f_{xxx} \right] \right. \\ &\quad \left. + \frac{1}{2} \left[ 2(\sqrt{\sigma}) f_x + 2 \frac{(\sqrt{\sigma})^3}{3!} f_{xxx} \right] \right\}^2 \\ &+ \frac{1}{2(\kappa+1)^2} \left\{ \kappa \left[ (\sqrt{\sigma}) f_x - \frac{(\sqrt{\sigma})^2}{2!} f_{xx} + \frac{(\sqrt{\sigma})^3}{3!} f_{xxx} \right] \right. \\ &\quad \left. + \frac{1}{2} \left[ 2(\sqrt{\sigma}) f_x + 2 \frac{(\sqrt{\sigma})^3}{3!} f_{xxx} \right] \right\}^2 \end{aligned}$$

Squaring these expression, retaining terms up to fourth order, and grouping terms in derivative order yield the following:

$$\begin{aligned} \bar{P}_{xx} &\approx (\sqrt{\sigma})^2 f_x^2 + \frac{(\kappa^2 + 2\kappa)(\sqrt{\sigma})^4}{(\kappa+1)^2} \frac{4}{4} f_{xx}^2 + \frac{2(\sqrt{\sigma})^4}{3!} f_x f_{xxx} \\ &= (\kappa+1) f_x^2 \sigma^2 + \left[ \frac{(\kappa^2 + 2\kappa)}{4} f_{xx}^2 + \frac{2(\kappa+1)^2}{3!} f_x f_{xxx} \right] \sigma^4 \end{aligned}$$

If the function  $f(x)$  were the instantiation of the state with the nonlinear system model, then this equation would represent the time propagation of the variance  $-\sigma^2$ , the predicted variance. Following this process for the other

terms, the predicated measurement variance and cross variance can be found and result in similar forms involving the measurement function  $-h(x)$ .

Continuing this example, by using the results from the preceding example, yields insight into the errors associated with the preceding predicted variance approximation. Returning to the preceding example, assuming the function  $y$  represents the instantiated state for time propagation, and if the deviation about the mean were Gaussian such that the expectation operators in this equation are related to the variance as

$$E[\delta x^2] = \sigma^2, \quad E[\delta x^4] = 3\sigma^4, \text{ etc.}$$

then the variance equation becomes

$$\begin{aligned} P_{yy} &= f_x^2 \sigma^2 + 3 * 2 \frac{1}{3!} f_{xxx} f_x \sigma^4 + 3 \frac{1}{2 * 2!} f_{xx}^2 \sigma^4 - \frac{1}{2! * 2!} f_{xx}^2 \sigma^4 + \dots \\ &= f_x^2 \sigma^2 + f_{xxx} f_x \sigma^4 + \frac{1}{2} f_{xx}^2 \sigma^4 + \dots \\ &= f_x^2 \sigma^2 + \left( \frac{1}{2} f_{xx}^2 + f_x f_{xxx} \right) \sigma^4 + \dots \end{aligned}$$

The difference between this form and that just given becomes approximately

$$\begin{aligned} \varepsilon P \equiv P_{yy} - \bar{P}_{xx} &= \left[ f_x^2 \sigma^2 + \left( \frac{1}{2} f_{xx}^2 + f_x f_{xxx} \right) \sigma^4 + \dots \right] \\ &\quad - \left\{ (\kappa + 1) f_x^2 \sigma^2 + \left[ \frac{(\kappa^2 + 2\kappa)}{4} f_{xx}^2 + \frac{2(\kappa + 1)^2}{3!} f_x f_{xxx} \right] \sigma^4 + \dots \right\} \\ &= -\kappa f_x^2 \sigma^2 + \left\{ \left[ \frac{1}{2} - \frac{(\kappa^2 + 2\kappa)}{4} \right] f_{xx}^2 + \left[ 1 - \frac{2(\kappa + 1)^2}{3!} \right] f_x f_{xxx} \right\} \sigma^4 \end{aligned}$$

The constant  $\kappa$  is chosen to minimize the magnitude of this difference. Mentioned in the literature [13] is the choice of setting this constant to zero. With this choice, the resulting variance difference becomes

$$\varepsilon P \approx \left( \frac{1}{2} f_{xx}^2 + \frac{2}{3} f_{xxx} f_x \right) \sigma^4$$

The predicted variance would approximate the true variance with an error including at least those terms remaining in this equation. The authors [12] suggest a specific value for the constant  $\kappa$  and/or additional constants to reduce this error. Current research is aimed at approximating this variance, using differences to approximate the derivatives, a selection of constants as in Table 8.2 plus others, or higher-order approximations for the derivatives in this expression.

### 8.7.2 Divided-Difference Filter

The following table summarizes the divided difference filter [15]. The algorithm shown in this table is expressed in matrix form; however, most references to this algorithm are expressed in matrix factored form including the one cited.

The algorithm's development is based an application of Stirling's interpolation formula [16]. A function is to be evaluated at an intermediate, interpolated, point between fixed points where its value has been already determined. The points where this function's values are specified are separated by fixed interval given by  $h$ . An approximation to this function of an intermediate point is evaluated by using this interpolation formula, which is given as

$$\begin{aligned} f(x + rh) = & f(x) + r \frac{1}{2} [f(x + h) - f(x - h)] \\ & + \frac{r^2}{2!} [f(x + h) - 2f(x) + f(x - h)] + \dots \end{aligned} \quad (8.90)$$

where for interpolation the variable  $r$  ranges between  $-1 < r < 1$ . For the divided-difference filter the variable  $r$  is defined by the ratio  $r \equiv \delta x/h$ . Substituting this expression into the first few terms in the interpolation formula yields

$$\begin{aligned} f(x + \delta x) = & f(x) + \left( \frac{\delta x}{h} \right) \frac{1}{2} [f(x + h) - f(x - h)] \\ & + \frac{1}{2!} \left( \frac{\delta x}{h} \right)^2 [f(x + h) - 2f(x) + f(x - h)] + \dots \end{aligned} \quad (8.91)$$

The fixed step intervals are redefined as dynamically scaled steps by replacing the step size variable  $h$  with  $h\sigma$ , where  $\sigma$  is a dynamic variable

$$\begin{aligned} f(x + \delta x) = & f(x) + \left( \frac{\delta x}{h\sigma} \right) \frac{1}{2} [f(x + h\sigma) - f(x - h\sigma)] \\ & + \frac{1}{2!} \left( \frac{\delta x}{h\sigma} \right)^2 [f(x + h\sigma) - 2f(x) + f(x - h\sigma)] + \dots \\ = & f(x) + \left( \frac{\delta x}{\sigma} \right) \frac{1}{2h} [f(x + h\sigma) - f(x - h\sigma)] \\ & + \left( \frac{\delta x}{\sigma} \right)^2 \frac{1}{2h^2} [f(x + h\sigma) - 2f(x) + f(x - h\sigma)] + \dots \end{aligned} \quad (8.92)$$

In the following, the variable  $\sigma$  will assume the characteristics of the root of the variance of the random variable  $\delta x$ . The variable  $(\delta x/\sigma)$  is assumed to be a normally distributed Gaussian random variable whose expectations are defined as

$$E\left[\left(\frac{\delta x}{\sigma}\right)^i\right] = 0 \quad \forall i - \text{odd powers} \quad (8.93)$$

and

$$E\left[\left(\frac{\delta x}{\sigma}\right)^i\right] = \sigma_i \quad \forall i - \text{even powers} \quad (8.94)$$

where, for Gaussian random variables,  $\sigma_i = 1^* 3^* \dots^* (i-1)$ , for example,

$$\sigma_2 = \sigma^2 \equiv 1, \quad \sigma_4 = 3\sigma^2 = 3, \quad \text{etc.} \quad (8.95)$$

The expectation of Eq. (8.92) becomes

$$\begin{aligned} E[y] &= E\left[f(x) + \left(\frac{\delta x}{\sigma}\right) \frac{1}{2h} [f(x+h\sigma) - f(x-h\sigma)] \right. \\ &\quad \left. + \frac{1}{2h^2} \left(\frac{\delta x}{\sigma}\right)^2 [f(x+h\sigma) - 2f(x) + f(x-h\sigma)] + \Lambda\right] \\ &= f(x) + E\left[\left(\frac{\delta x}{\sigma}\right)\right] \frac{1}{2h} [f(x+h\sigma) - f(x-h\sigma)] \\ &\quad + E\left[\left(\frac{\delta x}{\sigma}\right)^2\right] \frac{1}{2h^2} [f(x+h\sigma) - 2f(x) + f(x-h\sigma)] + \dots \\ &\approx f(x) + \frac{1}{2h^2} [f(x+h\sigma) - 2f(x) + f(x-h\sigma)] + \dots \end{aligned} \quad (8.96)$$

To order,  $E[(\delta x/\sigma)^3]$ . This equation can also be written in the form in Table 8.3 as

$$\begin{aligned} E[y] &= f(x) + \frac{1}{2h^2} [f(x+h\sigma) - 2f(x) + f(x-h\sigma)] \\ &= \left[1 - \frac{1}{h^2}\right] f(x) + \frac{1}{2h^2} [f(x+h\sigma) + f(x-h\sigma)] \\ &= \left[\frac{h^2 - 1}{h^2}\right] f(x) + \frac{1}{2h^2} [f(x+h\sigma) + f(x-h\sigma)] \end{aligned} \quad (8.97)$$

Table 8.3 Second-order divided-difference Kalman filter [15] (illustrated in matrix form)

Factor state covariance matrix	$\sqrt{P_i} \quad i = 1, 2, K, n$	ith row or column of square-root factor of $P$
State a priori update	$\bar{x} = \frac{(h^2 - n)}{h^2} f(x_0) + \frac{1}{2h^2} \sum_{i=1}^n [f(x_i + h\sqrt{P_i}) + f(x_i - h\sqrt{P_i})]$	Predicted mean from function's expectation based on Stirling interpolation formula to 2nd order
Covariance a priori update	$P_{xx} = \frac{1}{4h^2} \sum_{i=1}^n [f(x_i + h\sqrt{P_i}) - f(x_i - h\sqrt{P_i})]$ $* [f(x_i + h\sqrt{P_i}) - f(x_i - h\sqrt{P_i})]^T$ $+ \frac{(\sigma_4 - 1)}{4h^4} \sum_{i=1}^n [f(x_i + h\sqrt{P_i}) - 2f(x_0) + f(x_i - h\sqrt{P_i})]$ $* [f(x_i + h\sqrt{P_i}) - 2f(x_0) + f(x_i - h\sqrt{P_i})]^T \quad \{+Q\}$	Function error variance to 2nd order
Measurement a posteriori update	$\bar{y} = \frac{(h^2 - n)}{h^2} g(x_0) + \frac{1}{2h^2} \sum_{i=1}^n [g(x_i + h\sqrt{P_i}) + g(x_i - h\sqrt{P_i})]$ $P_{yy} = \frac{1}{4h^2} \sum_{i=1}^n [g(x_i + h\sqrt{P_i}) - g(x_i - h\sqrt{P_i})]^T$ $+ \frac{(\sigma_4 - 1)}{4h^4} \sum_{i=1}^n [g(x_i + h\sqrt{P_i}) - 2g(x_0) + g(x_i - h\sqrt{P_i})]$ $* [g(x_i + h\sqrt{P_i}) - 2g(x_0) + g(x_i - h\sqrt{P_i})]^T \quad \{+R\}$	Same as predicted mean Same as predicted variance
Innovation covariance a posteriori update	$P_{xy} = \frac{1}{2h} \sum_{i=1}^n \sqrt{P_i} [g(x_i + h\sqrt{P_i}) - g(x_i - h\sqrt{P_i})]$	Mean and measurement prediction variance
Cross covariance	$K = P_{xy} P_{yy}^{-1}$	Return to top
Kalman filter equations	$\hat{x} = \bar{x} + K(y - \bar{y})$ $P = P_{xx} - K P_{yy} K^T$	

The difference between the function and its expectation becomes

$$\begin{aligned}y - E[y] &= f(x) + \left(\frac{\delta x}{\sigma}\right) \frac{1}{2h} [f(x + h\sigma) - f(x - h\sigma)] \\&\quad + \frac{1}{2h^2} \left(\frac{\delta x}{\sigma}\right)^2 [f(x + h\sigma) - 2f(x) + f(x - h\sigma)] + \dots \\&\quad - f(x) - \frac{1}{2h^2} [f(x + h\sigma) - 2f(x) + f(x - h\sigma)]\end{aligned}$$

or

$$\begin{aligned}y - E[y] &= \left(\frac{\delta x}{\sigma}\right) \frac{1}{2h} [f(x + h\sigma) - f(x - h\sigma)] \\&\quad + \left[ \left(\frac{\delta x}{\sigma}\right)^2 - 1 \right] \frac{1}{2h^2} [f(x + h\sigma) - 2f(x) + f(x - h\sigma)] + \dots \quad (8.98)\end{aligned}$$

Defining temporary variables

$$a \equiv \frac{1}{2h} [f(x + h\sigma) - f(x - h\sigma)]$$

and

$$b \equiv \frac{1}{2h^2} [f(x + h\sigma) - 2f(x) + f(x - h\sigma)]$$

The preceding difference can be written, including the first two terms, as

$$y - E[y] \approx \left(\frac{\delta x}{\sigma}\right) a + \left[ \left(\frac{\delta x}{\sigma}\right)^2 - 1^2 \right] b$$

The variance of this difference becomes

$$\begin{aligned}E[(y - E[y])^2] &= E\left(\left\{\left(\frac{\delta x}{\sigma}\right)a + \left[\left(\frac{\delta x}{\sigma}\right)^2 - 1\right]b\right\}^2\right) \\&= E\left\{\left(\frac{\delta x}{\sigma}\right)^2 a^2 + 2\left(\frac{\delta x}{\sigma}\right)\left[\left(\frac{\delta x}{\sigma}\right)^2 - 1\right]ab + \left[\left(\frac{\delta x}{\sigma}\right)^2 - 1\right]^2 b^2\right\}\end{aligned}$$

$$\begin{aligned}
&\approx E\left[\left(\frac{\delta x}{\sigma}\right)^2\right]a^2 + E\left\{\left[\left(\frac{\delta x}{\sigma}\right)^2 - 1\right]^2\right\}b^2 \\
&= E\left[\left(\frac{\delta x}{\sigma}\right)^2\right]a^2 + \left\{E\left[\left(\frac{\delta x}{\sigma}\right)^4\right] - 2E\left[\left(\frac{\delta x}{\sigma}\right)^2\right] + 1\right\}b^2 \\
&= a^2 + (\sigma_4 - 1)b^2
\end{aligned}$$

Or in terms of the interpolation formula's differences

$$\begin{aligned}
E[(y - E[y])^2] &= \frac{1}{4h^2}[f(x + h\sigma) - f(x - h\sigma)]^2 \\
&+ \frac{(\sigma_4 - 1)}{4h^4}[f(x + h\sigma) - 2f(x) + f(x - h\sigma)]^2 \quad (8.99)
\end{aligned}$$

This expression is applied in forming both the state propagation and measurement variances. In the former, the symbol  $f(x)$  is the state propagation function, and, for the latter, the function  $g(x)$  is used in its place for the measurement. In its matrix form, this expression's bracketed squared terms are sums of vector outer products as shown in Table 8.3.

The first of the differences in Eq. (8.99) is associated with the divided-difference filter variance, and, with the addition of the second term, a second-order divided-difference filter is formed.

The cross covariance between the state and measurement is required for the algorithm and is formed next. The expectation between the state  $x$ , and measurement assumed by the variable  $y$ , is defined as

$$P_{xy} = E[(x - E[x])(y - E[y])] \quad (8.100)$$

Referring to the preceding definition of the expansion,  $x - E[x] = \delta x = (\delta x/\sigma)\sigma$ . Using the function defined as the measurement  $g(x)$ , the difference between the measurement and its expectation is written as

$$\begin{aligned}
y - E[y] &= \left(\frac{\delta x}{\sigma}\right)\frac{1}{2h}[g(x + h\sigma) - g(x - h\sigma)] \\
&+ \left[\left(\frac{\delta x}{\sigma}\right)^2 - 1\right]\frac{1}{2h^2}[g(x + h\sigma) - 2g(x) + g(x - h\sigma)] \quad (8.101)
\end{aligned}$$

The cross variance becomes

$$\begin{aligned}
 P_{xy} &= \sigma E \left[ \left( \frac{\delta x}{\sigma} \right)^2 \frac{1}{2h} [g(x + h\sigma) - g(x - h\sigma)] \right. \\
 &\quad \left. + \left( \frac{\delta x}{\sigma} \right) \left[ \left( \frac{\delta x}{\sigma} \right)^2 - 1 \right] \frac{1}{2h^2} [g(x + h\sigma) - 2g(x) + g(x - h\sigma)] \right] \\
 &\approx \sigma E \left[ \left( \frac{\delta x}{\sigma} \right)^2 \right] \frac{1}{2h} [g(x + h\sigma) - g(x - h\sigma)] \\
 &= \sigma \sigma_2 \frac{1}{2h} [g(x + h\sigma) - g(x - h\sigma)]
 \end{aligned} \tag{8.102}$$

Or, because  $\sigma_2 = \sigma^2 = 1$ ,

$$P_{xy} = \sigma \left\{ \frac{1}{2h} [g(x + h\sigma) - g(x - h\sigma)] \right\} \tag{8.103}$$

This difference is associated with both the divided-difference and the second-order divided-difference filters. The extension of these scalar forms to vector form is straightforward.

## 8.8 Chapter Summary

The approach of Sage [2] was followed in developing the least-squares estimation algorithm. Initially, the algorithm was developed for constant systems with no dynamics. Then, proceeding from the WLS estimation algorithm, a RWLS estimation algorithm was developed. Time-varying system dynamics were incorporated with the algorithm's development, again using the RWLS approach. This final least-squares step resulted in a linear discrete estimation algorithm similar in form to the standard Kalman filter algorithm.

The Kalman filter algorithm was derived again by establishing a minimum-variance linear estimator. The linear estimator form, motivated from the result obtained from the least-squares approach, was assumed; however, in this derivation it was for a fully dynamic system model. The resulting estimator is the discrete Kalman filter algorithm in Joseph form. The two approaches, least squares and minimum-variance estimators, were demonstrated to result in the same estimation algorithm.

Starting with the standard Kalman filter measurement update form, the  $U-D$  factored form developed by Bierman was developed for scalar measurement update equations.

The different forms for the Kalman filter algorithm, standard, Joseph, and  $U-D$  factored, are available to the designer/integrator. The decision on which one to use can be evaluated, to some degree, by the computational impacts of its implementation. The problems in this chapter illustrate their relative computational impacts.

Additional topics were presented, including algorithm modifications for summed measurements, a method for optimally combining filter estimates from two filters, and a discussion concerning two algorithm forms for derivative-free filtering.

### Problems

- 8.1** Consider the alternate gain form. Obtain the gain expression, originally given in Eq. (8.11) but stated in Example 8.1 as  $\mathbf{k}_m = P_{m+1}\mathbf{h}r^{-1}$ . From Eq. (8.11), the gain is defined as

$$\mathbf{k}_m = P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1}$$

Use the identity

$$P_{m+1} P_{m+1}^{-1} = I$$

and premultiply the gain equation by this identity, which represents no change because the multiplication is by the identity matrix  $I$ , to obtain

$$\mathbf{k}_m = P_{m+1} P_{m+1}^{-1} P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1}$$

Use Eq. (8.8) to obtain the alternate form for the gain equation as

$$\begin{aligned} \mathbf{k}_m &= P_{m+1} [P_m^{-1} + \mathbf{h}r^{-1}\mathbf{h}] P_m \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \\ &= P_{m+1} [I + \mathbf{h}r^{-1}\mathbf{h}^T P_m] \mathbf{h} (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \\ &= P_{m+1} \mathbf{h} [1 + r^{-1} \mathbf{h}^T P_m \mathbf{h}] (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \\ &= P_{m+1} \mathbf{h} \left[ \frac{r + \mathbf{h}^T P_m \mathbf{h}}{r} \right] (\mathbf{h}^T P_m \mathbf{h} + r)^{-1} \\ &= P_{m+1} \mathbf{h} r^{-1} \end{aligned}$$

- 8.2** Consider the number of numerical operations for Kalman gain calculation. For a single scalar measurement and a three-state filter state vector, confirm that  $n^2 + n$  multiplications,  $n^2$  additions, and one division are required to form the Kalman gain. Recall that the Kalman gain is computed by

$$\mathbf{k} = (\mathbf{h}^T \mathbf{P} \mathbf{h} + r)^{-1} \mathbf{P} \mathbf{h}$$

For the product  $\mathbf{P} \mathbf{h}$ , taking advantage of the symmetry of the matrix  $\mathbf{P}$ , the following is obtained:

$$\mathbf{P} \mathbf{h} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{12} & p_{22} & p_{23} \\ p_{13} & p_{23} & p_{33} \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} h_1 p_{11} + h_2 p_{12} + h_3 p_{13} \\ h_1 p_{12} + h_2 p_{22} + h_3 p_{23} \\ h_1 p_{13} + h_2 p_{23} + h_3 p_{33} \end{bmatrix}$$

where there are  $n^2$  multiplications and  $n(n - 1)$  additions. For the product  $\mathbf{h}^T \mathbf{P} \mathbf{h}$

$$\begin{aligned}\mathbf{h}^T \mathbf{P} \mathbf{h} &= h_1(h_1 p_{11} + h_2 p_{12} + h_3 p_{13}) \\ &\quad + h_2(h_1 p_{12} + h_2 p_{22} + h_3 p_{23}) \\ &\quad + h_3(h_1 p_{13} + h_2 p_{23} + h_3 p_{33})\end{aligned}$$

there are an additional  $n$  multiplications and  $n - 1$  additions. The sum of  $\mathbf{h}^T \mathbf{P} \mathbf{h}$  and  $r$  requires one additional addition. One division is required to form  $(\mathbf{h}^T \mathbf{P} \mathbf{h} + r)^{-1}$  which, when multiplied to each of the preceding first vector's terms, requires  $n$  additional multiplications.

- 8.3** Consider the number of numerical operations for standard and Joseph forms. Assuming common computations for the Kalman gain in the preceding problem for both the standard and Joseph Kalman filter forms, and again using a single scalar measurement with a three-state filter state vector, confirm that the total number of operations for the gain and covariance measurement update are as follows:

Operation	Standard	Joseph
Multiplications	$\frac{3n^2 + 5n}{2}$	$4n^2 + 4n$
Additions	$\frac{3n^2 + n}{2}$	$\frac{7n^2 + n}{2}$
Divisions	1	1

*Hint:* From Bierman [3], rewrite the standard Kalman filter measurement update equation as

$$\begin{aligned}\hat{\mathbf{P}} &= [\mathbf{I} - \mathbf{k} \mathbf{h}^T] \tilde{\mathbf{P}} \\ &= \tilde{\mathbf{P}} - \mathbf{k} (\tilde{\mathbf{P}} \mathbf{h})^T\end{aligned}$$

where a newly defined variable  $\mathbf{v}$  has been formed in the preceding example as

$$\mathbf{v} \equiv \tilde{\mathbf{P}} \mathbf{h}$$

Then, the product on the RHS becomes

$$\mathbf{k} \mathbf{v}^T = \begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} = \begin{bmatrix} k_1 v_1 & k_1 v_2 & k_1 v_3 \\ k_2 v_1 & k_2 v_2 & k_2 v_3 \\ k_3 v_1 & k_3 v_2 & k_3 v_3 \end{bmatrix}$$

which must be symmetric, resulting in  $n(n + 1)/2$  multiplications. Subtraction of this result from the  $\tilde{P}$  matrix, again taking advantage of symmetry, requires  $n(n + 1)/2$  additions. This result for the standard Kalman measurement update, added to that for computing the Kalman gain, results in the values shown in the preceding table.

- 8.4** Consider the number of numerical operations for  $U-D$  factored form. Again using a single scalar measurement with a three-state filter state vector, confirm that the number of total operations for the Kalman gain and covariance measurement update are as follows:

Multiplications:

$$\frac{3n^2 + 9n}{2}$$

Additions:

$$\frac{3n^2 - n}{2}$$

and

Divisions:

$$n$$

*Hint:* Use the results presented in Example 8.3.

- 8.5** Residual editing in a measurement update is presented in this problem. A pre-measurement assessment of measurement data might be required to prevent the results obtained from the Kalman filter to be corrupted as the result of using poor quality data.

Form the measurement residual as

$$\boldsymbol{v}_{k+1} = \mathbf{z}_{k+1} - H_{k+1}\tilde{\mathbf{x}}_{k+1}$$

using the equation for the measurement

$$\mathbf{z}_{k+1} = H_{k+1}\mathbf{x}_{k+1} + \boldsymbol{v}_{k+1}$$

and the definition of the estimation error

$$\tilde{\boldsymbol{\varepsilon}}_{k+1} = \mathbf{x}_{k+1} - \tilde{\mathbf{x}}_{k+1}$$

Express the measurement residual as

$$\mathbf{v}_{k+1} = H_{k+1}\tilde{\boldsymbol{\epsilon}}_{k+1} + \mathbf{v}_{k+1}$$

Obtain the following form for the residual covariance:

$$E[\mathbf{v}_{k+1}\mathbf{v}_{k+1}^T] = H_{k+1}\tilde{P}_{k+1}H_{k+1}^T + R_{k+1}$$

*Hint:* The residual covariance is formed by

$$\begin{aligned} E[\mathbf{v}_{k+1}\mathbf{v}_{k+1}^T] &= E[(H_{k+1}\tilde{\boldsymbol{\epsilon}}_{k+1} + \mathbf{v}_{k+1})(H_{k+1}\tilde{\boldsymbol{\epsilon}}_{k+1} + \mathbf{v}_{k+1})^T] \\ &= H_{k+1}E[\tilde{\boldsymbol{\epsilon}}_{k+1}\tilde{\boldsymbol{\epsilon}}_{k+1}^T]H_{k+1}^T + H_{k+1}E[\tilde{\boldsymbol{\epsilon}}_{k+1}\mathbf{v}_{k+1}^T] \\ &\quad + E[\mathbf{v}_{k+1}\tilde{\boldsymbol{\epsilon}}_{k+1}^T]H_{k+1}^T + E[\mathbf{v}_{k+1}\mathbf{v}_{k+1}^T] \end{aligned}$$

where, in proceeding to the residual covariance equation, the following two terms are assumed to be zero, for example,  $E[\tilde{\boldsymbol{\epsilon}}_{k+1}\mathbf{v}_{k+1}^T] = E[\mathbf{v}_{k+1}\tilde{\boldsymbol{\epsilon}}_{k+1}^T] = [0]$ .

The residual edit process is used to exclude measurements that deviate from the expected measurement values as computed from the filter state estimates. Treating a scalar measurement, rationalize the use of the following test made prior to proceeding with the measurement update:

$$|z - \mathbf{h}^T\hat{\mathbf{x}}| \gg \sqrt{\mathbf{h}^T\tilde{P}\mathbf{h} + r}$$

That is, if the magnitude of the residual is much greater than the corresponding residual variance (square root), then the measurement update associated with that measurement data should be skipped. A multiple of the residual variance, ranging from three to six, is typically used, so that the test becomes whether the residual magnitude is greater than this multiple of the residual variance.

- 8.6** This problem examines the stability for the continuous form of the Kalman filter. A linear continuous system is described by the equations

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= F\hat{\mathbf{x}} + w \\ z &= H\hat{\mathbf{x}} + v \end{aligned}$$

where

$$E[ww^T] = Q \quad \text{and} \quad E[vv^T] = R$$

and the continuous Kalman filter is given by

$$\dot{\hat{\mathbf{x}}} = F\hat{\mathbf{x}} + K(z - H\hat{\mathbf{x}})$$

with gain

$$K = PH^TR^{-1}$$

Define the estimation error as

$$\varepsilon \equiv x - \hat{x}$$

with associated estimation error covariance defined as

$$P \equiv E[\varepsilon \varepsilon^T]$$

The differential equation for this matrix is given as

$$\dot{P} = FP + PF^T + Q - PH^T R^{-1} HP$$

Stability of this continuous filter form is studied by examining the dynamic equation for the filter's estimation error. Form the following differential equation for the estimation error:

$$\begin{aligned}\dot{\varepsilon} &= \dot{x} - \dot{\hat{x}} = Fx + w - F\hat{x} - K[(Hx + v) - H\hat{x}] \\ &= F(x - \hat{x}) - KH(x - \hat{x}) + w - Kv \\ &= (F - KH)\varepsilon + w - Kv\end{aligned}$$

This differential equation's stability is governed by the matrix  $(F - KH)$ . Assess the stability of this equation by Lapunov techniques. The scalar function Lapunov function is defined as

$$V \equiv \varepsilon^T P^{-1} \varepsilon$$

Asymptotic stability is guaranteed if  $\dot{V} \leq 0$ . Form the derivative of this function by first establishing the derivative of the inverse of the estimation error covariance matrix. Form the identity

$$PP^{-1} = I$$

Take its derivative yielding

$$\frac{d}{dt}(PP^{-1}) = \dot{P}P^{-1} + P\dot{P}^{-1} = [0]$$

Then solve for the inverse differential equation

$$\dot{P}^{-1} = -P^{-1}\dot{P}P^{-1}$$

Substitute the preceding differential equation for the covariance matrix to obtain

$$\begin{aligned}\dot{P}^{-1} &= -P^{-1}[FP + PF^T + Q - PH^T R^{-1} HP]P^{-1} \\ &= -P^{-1}F + F^T P^{-1} + P^{-1}Q P^{-1} - H^T R^{-1} H\end{aligned}$$

Assess stability about an assumed steady state:

$$\dot{P}^{-1} = [0]$$

Form the derivative of the Lapunov function in the following sequence of steps:

$$\begin{aligned}\dot{V} &\equiv \dot{\varepsilon}^T P^{-1} \varepsilon + \varepsilon^T \dot{P}^{-1} \varepsilon + \varepsilon^T P^{-1} \dot{\varepsilon} \\&= 2\varepsilon^T P^{-1} \dot{\varepsilon} \\&= 2\varepsilon^T P^{-1} (F - KH) \varepsilon \\&= 2\varepsilon^T (F - PH^T R^{-1} H) \varepsilon \\&= 2\varepsilon^T (P^{-1} F - H^T R^{-1} H) \varepsilon \\&= \varepsilon^T [(P^{-1} F - H^T R^{-1} H) + (P^{-1} F - H^T R^{-1} H)^T] \varepsilon \\&= \varepsilon^T [(-F^T P^{-1} - P^{-1} Q P^{-1}) + (P^{-1} F - H^T R^{-1} H)^T] \varepsilon \\&= -\varepsilon^T [P^{-1} Q P^{-1} + H^T R^{-1} H^T] \varepsilon\end{aligned}$$

Therefore, the system is asymptotically stable if the matrix  $[P^{-1} Q P^{-1} + H^T R^{-1} H^T]$  is positive definite.

- 8.7** Examine the response of a continuous Kalman (Kalman [17]) filter using a scalar example. Consider the scalar time-invariant linear system

$$\begin{aligned}\dot{x} &= \alpha x + w \\z &= x + v\end{aligned}$$

The estimation error variance dynamic equation is given as

$$\dot{\sigma} = 2\alpha\sigma + q - \frac{\sigma^2}{r}$$

Form its integral

$$\int_{t_0}^t dt = \int_{\sigma_0}^{\sigma} \frac{d\sigma}{2\alpha\sigma + q - \sigma^2/r}$$

The solution of an integral of this form can be found to be

$$\int \frac{dx}{X} = \frac{1}{\sqrt{-q}} \ln \left[ \frac{2cx + b - \sqrt{-q}}{2cx + b + \sqrt{-q}} \right], \quad X = a + bx + cx^2, \quad q = 4ac - b^2$$

Identifying the terms in this solution corresponding to those in the integral, find the solution here:

$$\frac{\sigma}{r} = \frac{[\alpha + (\alpha^2 + q/r)^{1/2}] \left\{ -\frac{\sigma_0}{r} + \alpha - (\alpha^2 + q/r)^{1/2} \right\} e^{2(\alpha^2 + q/r)^{1/2}(t-t_0)}}{\frac{-[\alpha - (\alpha^2 + q/r)^{1/2}] \left\{ -\frac{\sigma_0}{r} + \alpha + (\alpha^2 + q/r)^{1/2} \right\}}{\left\{ -\frac{\sigma_0}{r} + \alpha - (\alpha^2 + q/r)^{1/2} \right\} e^{2(\alpha^2 + q/r)^{1/2}(t-t_0)}} - \left\{ -\frac{\sigma_0}{r} + \alpha + (\alpha^2 + q/r)^{1/2} \right\}}$$

The steady-state solution can be found to be

$$\lim_{t \rightarrow \infty} \left( \frac{\sigma}{r} \right) = r[\alpha + (\alpha^2 + q/r)^{1/2}]$$

if the system dynamics are such that the following relationship holds,

$$\alpha \ll q/r$$

then the solution can be approximated by

$$\frac{\sigma}{r} \approx (q/r)^{1/2} \left[ \frac{ae^{2(q/r)^{1/2}\Delta t} + b}{ae^{2(q/r)^{1/2}\Delta t} - b} \right]$$

This approximate solution suggests a system time constant formed by the root of the ratio of process to measurement noise

$$\frac{1}{\tau} \sim \left( \frac{q}{r} \right)^{1/2}$$

which further suggests that the greater the system noise power to observation noise, the slower the system responses.

- 8.8** The  $U-D$  factored form's time update is considered in this problem. Rewrite Eq. (8.62) in the following form:

$$\tilde{U}\tilde{D}\tilde{U}^T = \hat{\Phi}\hat{U}\hat{D}\hat{U}^T\hat{\Phi}^T + \Gamma Q \Gamma^T$$

Continue to use the  $3 \times 3$  example presented in Example 8.3 and form the following:

$$\begin{aligned} \hat{\Phi}\hat{U} &= \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{bmatrix} \begin{bmatrix} 1 & \hat{u}_{12} & \hat{u}_{13} \\ 0 & 1 & \hat{u}_{23} \\ 0 & 0 & 1 \end{bmatrix} \equiv A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \\ &\equiv \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \mathbf{a}_3^T \end{bmatrix} \end{aligned}$$

Assume that the last term in the preceding equation can be written as the diagonal matrix

$$\Gamma Q \Gamma^T \equiv \begin{bmatrix} q_1 & 0 & 0 \\ 0 & q_2 & 0 \\ 0 & 0 & q_3 \end{bmatrix}$$

Rewrite the second term in this equation, using the definition of the preceding  $A$  matrix, as

$$A \hat{D} A^T = \begin{bmatrix} \mathbf{a}_1^T \hat{D} \mathbf{a}_1 & \mathbf{a}_1^T \hat{D} \mathbf{a}_2 & \mathbf{a}_1^T \hat{D} \mathbf{a}_3 \\ \mathbf{a}_2^T \hat{D} \mathbf{a}_1 & \mathbf{a}_2^T \hat{D} \mathbf{a}_2 & \mathbf{a}_2^T \hat{D} \mathbf{a}_3 \\ \mathbf{a}_3^T \hat{D} \mathbf{a}_1 & \mathbf{a}_3^T \hat{D} \mathbf{a}_2 & \mathbf{a}_3^T \hat{D} \mathbf{a}_3 \end{bmatrix}$$

As in Example 8.3, start in the lower right-hand corner, equate matrix elements for both sides of the preceding equation, and obtain the following:

$$\begin{aligned} \tilde{d}_3 &= \mathbf{a}_3^T \hat{D} \mathbf{a}_3 + q_3 \\ \tilde{u}_{23} &= \frac{\mathbf{a}_2^T \hat{D} \mathbf{a}_3}{\tilde{d}_3} \\ \tilde{u}_{13} &= \frac{\mathbf{a}_1^T \hat{D} \mathbf{a}_3}{\tilde{d}_3} \\ \tilde{d}_2 &= \mathbf{a}_2^T \hat{D} \mathbf{a}_2 + q_2 - \mathbf{a}_2^T \hat{D} \mathbf{a}_3 \tilde{u}_{23} \\ \tilde{u}_{12} &= \frac{(\mathbf{a}_1^T \hat{D} \mathbf{a}_2 - \mathbf{a}_1^T \hat{D} \mathbf{a}_3 \tilde{u}_{23})}{\tilde{d}_2} \\ \tilde{d}_1 &= \mathbf{a}_1^T \hat{D} \mathbf{a}_1 + q_1 - (\mathbf{a}_1^T \hat{D} \mathbf{a}_2 - \mathbf{a}_1^T \hat{D} \mathbf{a}_3 \tilde{u}_{23}) \tilde{u}_{12} - \mathbf{a}_1^T \hat{D} \mathbf{a}_3 \tilde{u}_{13} \end{aligned}$$

Note: The  $\mathbf{a}_i$  terms contain the measurement updated  $\hat{u}_{ij}$  terms. Compare these terms to the time update algorithm in [3].

- 8.9** Consider a conditional mean estimate [18] based on two jointly Gaussian vectors. The vector

$$\mathbf{Z} = \begin{bmatrix} X \\ Y \end{bmatrix}$$

is jointly Gaussian with mean and covariance

$$\mathbf{m} = \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix} \quad \Sigma = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}$$

and

$$\Sigma \quad \text{and} \quad \Sigma_{yy}$$

are nonsingular. Show that the vector  $X$  is conditionally Gaussian. The conditional probability density function is defined as

$$p_{X|Y} \equiv \frac{p_{XY}(x,y)}{p_Y(y)} = \frac{1}{(2\pi)^{N/2}} \frac{|\Sigma_{yy}|^{1/2}}{|\Sigma|^{1/2}} \frac{\exp \left\{ -\frac{1}{2} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} \right\}}{\exp \left\{ -\frac{1}{2} [y - \bar{y}]^T \Sigma_{yy}^{-1} [y - \bar{y}] \right\}}$$

Form the products using the results from Problem 2.14:

$$\begin{aligned} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} &= \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix}^T \begin{bmatrix} I & 0 \\ -\Sigma_{yy}^{-1} \Sigma_{xy}^T & I \end{bmatrix} \\ &\quad \times \begin{bmatrix} (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} & 0 \\ 0 & \Sigma_{yy}^{-1} \end{bmatrix} \\ &\quad \times \begin{bmatrix} I & -\Sigma_{xy} \Sigma_{yy}^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} \\ &= \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix}^T \begin{bmatrix} (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \\ -\Sigma_{yy}^{-1} \Sigma_{xy}^T (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \end{bmatrix} \\ &\quad \times \begin{bmatrix} -(\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \Sigma_{xy} \Sigma_{yy}^{-1} \\ \Sigma_{yy}^{-1} + \Sigma_{yy}^{-1} \Sigma_{xy}^T (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \Sigma_{xy} \Sigma_{yy}^{-1} \end{bmatrix} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} \\ &= (x - \bar{x})^T (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} (x - \bar{x}) \\ &\quad - (x - \bar{x})^T (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \Sigma_{xy} \Sigma_{yy}^{-1} (y - \bar{y}) \\ &\quad - (y - \bar{y})^T \Sigma_{yy}^{-1} \Sigma_{xy}^T \left( \Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx} \right)^{-1} (x - \bar{x}) \\ &\quad + (y - \bar{y})^T \left[ \Sigma_{yy}^{-1} + \Sigma_{yy}^{-1} \Sigma_{xy}^T (\Sigma_{xx} \right. \\ &\quad \left. - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \Sigma_{xy} \Sigma_{yy}^{-1} \right] (y - \bar{y}) \\ &= (x - \bar{x})^T (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} (x - \bar{x}) \\ &\quad - 2(x - \bar{x})^T (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \Sigma_{xy} \Sigma_{yy}^{-1} (y - \bar{y}) \\ &\quad + (y - \bar{y})^T \Sigma_{yy}^{-1} (y - \bar{y}) + \underline{(y - \bar{y})^T} \\ &\quad \times \underline{\Sigma_{yy}^{-1} \Sigma_{xy}^T (\Sigma_{xx} - \Sigma_{xy} \Sigma_{yy}^{-1} \Sigma_{yx})^{-1} \Sigma_{xy} \Sigma_{yy}^{-1} (y - \bar{y})} \end{aligned}$$

Define the estimate  $\hat{x}$  as

$$\hat{x} \equiv \bar{x} + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \bar{y})$$

resulting in the following equality of the preceding underlined terms:

$$\Sigma_{xy}\Sigma_{yy}^{-1}(y - \bar{y}) = \hat{x} - \bar{x}$$

Then, show that the preceding product becomes

$$\begin{aligned} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} &= (x - \bar{x})^T (\Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx})^{-1} \\ &\quad \times (x - \bar{x}) - 2(x - \bar{x})^T (\Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx})^{-1}(\hat{x} - \bar{x}) \\ &\quad + (y - \bar{y})^T \Sigma_{yy}^{-1} (y - \bar{y}) + (\hat{x} - \bar{x})^T \\ &\quad \times (\Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx})^{-1}(\hat{x} - \bar{x}) \end{aligned}$$

Use the following to simplify this product

$$(x - \bar{x})^T A(x - \bar{x}) - 2(x - \bar{x})^T A(\hat{x} - \bar{x}) + (\hat{x} - \bar{x})^T A(\hat{x} - \bar{x}) = (x - \hat{x})^T A(x - \hat{x})$$

to obtain

$$\begin{aligned} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix}^T \Sigma^{-1} \begin{bmatrix} x - \bar{x} \\ y - \bar{y} \end{bmatrix} &= (x - \hat{x})^T (\Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx})^{-1}(x - \hat{x}) \\ &\quad + (y - \bar{y})^T \Sigma_{yy}^{-1} (y - \bar{y}) \end{aligned}$$

Therefore,  $x$ , conditioned on  $y$ , has conditional mean  $\bar{x} + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \bar{y})$  and conditional covariance  $\Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}$ .

- 8.10** This exercise demonstrates several relationships between the UKF algorithm and a time-invariance linear system. Consider the linear system

$$x_{i+1} = \Phi x_i \Rightarrow \tilde{x} = \Phi x \quad \text{and} \quad y_i = Hx_i$$

Form the following:

Predicted mean:

$$\begin{aligned} \bar{x} &= \sum_{i=0}^{2n} W_i \tilde{\chi}_i = W_0 \tilde{\chi}_0 + W_1 \tilde{\chi}_1 + W_2 \tilde{\chi}_2 + \cdots + W_n \tilde{\chi}_n \\ &\quad + W_{n+1} \tilde{\chi}_{n+1} + W_{n+2} \tilde{\chi}_{n+2} + \cdots + W_{2n} \tilde{\chi}_{2n} \end{aligned}$$

$$\begin{aligned}
&= \frac{\kappa}{n+\kappa} \tilde{\chi}_0 + \frac{1}{2(n+\kappa)} \{ \tilde{\chi}_1 + \tilde{\chi}_2 + \cdots + \tilde{\chi}_n \} \\
&\quad + \frac{1}{2(n+\kappa)} \{ \tilde{\chi}_{n+1} + \tilde{\chi}_{n+2} + \cdots + \tilde{\chi}_{2n} \} \\
&= \frac{\kappa}{n+\kappa} \Phi x + \frac{1}{2(n+\kappa)} \{ \Phi(x + \sqrt{1}) + \Phi(x + \sqrt{2}) + \cdots \} \\
&\quad + \frac{1}{2(n+\kappa)} \{ \Phi(x - \sqrt{1}) + \Phi(x - \sqrt{2}) + \cdots \} \\
&= \frac{\kappa}{n+\kappa} \Phi x + \frac{2n}{2(n+\kappa)} \Phi x \\
&= \Phi x
\end{aligned}$$

Predicted covariance:

$$\begin{aligned}
\bar{P}_{xx} &= \sum_{i=0}^{2n} W_i (\tilde{\chi}_i - \bar{x})(\tilde{\chi}_i - \bar{x})^T = W_0 (\tilde{\chi}_0 - \bar{x})(\tilde{\chi}_0 - \bar{x})^T \\
&\quad + W_1 (\tilde{\chi}_1 - \bar{x})(\tilde{\chi}_1 - \bar{x})^T + \cdots \\
&\quad + W_{n+1} (\tilde{\chi}_{n+1} - \bar{x})(\tilde{\chi}_{n+1} - \bar{x})^T + \cdots \\
&= \frac{\kappa}{n+\kappa} (\Phi x - \Phi x)(\Phi x - \Phi x)^T \\
&\quad + \frac{1}{2(n+\kappa)} \{ [\Phi(x + \sqrt{1}) - \Phi x][\Phi(x + \sqrt{1}) - \Phi x]^T \\
&\quad + [\Phi(x + \sqrt{2}) - \Phi x][\Phi(x + \sqrt{2}) - \Phi x]^T + \cdots \} \\
&\quad + \frac{1}{2(n+\kappa)} \{ [\Phi(x - \sqrt{1}) - \Phi x][\Phi(x - \sqrt{1}) - \Phi x]^T \\
&\quad + [\Phi(x - \sqrt{2}) - \Phi x][\Phi(x - \sqrt{2}) - \Phi x]^T + \cdots \} \\
&= \frac{1}{2(n+\kappa)} \{ \Phi \sqrt{1} \sqrt{1}^T \Phi^T + \Phi \sqrt{2} \sqrt{2}^T \Phi^T + \cdots \} \\
&\quad + \frac{1}{2(n+\kappa)} \{ \Phi \sqrt{1} \sqrt{1}^T \Phi^T + \Phi \sqrt{2} \sqrt{2}^T \Phi^T + \cdots \} \\
&= \frac{2}{2(n+\kappa)} \Phi \{ \sqrt{1} \sqrt{1}^T + \sqrt{2} \sqrt{2}^T + \cdots + \sqrt{n} \sqrt{n}^T \} \Phi^T = \Phi P \Phi^T
\end{aligned}$$

Mean measurement:

$$\begin{aligned}
 \bar{y} &= \sum_{i=0}^{2n} W_i Y_i = W_0 Y_0 + W_1 Y_1 + W_2 Y_2 + \cdots + W_n Y_n + W_{n+1} Y_{n+1} \\
 &\quad + W_{n+2} Y_{n+2} + \cdots + W_{2n} Y_{2n} \\
 &= \frac{\kappa}{n+\kappa} Hx + \frac{1}{2(n+\kappa)} \{H(x + \sqrt{1}) + H(x + \sqrt{2}) + \cdots\} \\
 &\quad + \frac{1}{2(n+\kappa)} \{H(x - \sqrt{1}) + H(x - \sqrt{2}) + \cdots\} \\
 &= \frac{\kappa}{n+\kappa} Hx + \frac{2n}{2(n+\kappa)} Hx \\
 &= Hx
 \end{aligned}$$

Residual variance:

$$\begin{aligned}
 P_{yy} &= \sum_{i=0}^{2n} W_i (Y_i - \bar{y})(Y_i - \bar{y})^T = W_0 (Y_0 - \bar{y})(Y_0 - \bar{y})^T \\
 &\quad + W_1 (Y_1 - \bar{y})(Y_1 - \bar{y})^T + \cdots + W_{n+1} (Y_{n+1} - \bar{y})(Y_{n+1} - \bar{y})^T + \cdots \\
 &= \frac{\kappa}{n+\kappa} (Hx - Hx)(Hx - Hx)^T \\
 &\quad + \frac{1}{2(n+\kappa)} \{[H(x + \sqrt{1}) - Hx][H(x + \sqrt{1}) - Hx]^T \\
 &\quad + [H(x + \sqrt{2}) - Hx][H(x + \sqrt{2}) - Hx]^T + \cdots\} \\
 &\quad + \frac{1}{2(n+\kappa)} \{[H(x - \sqrt{1}) - Hx][H(x - \sqrt{1}) - Hx]^T \\
 &\quad + [H(x - \sqrt{2}) - Hx][H(x - \sqrt{2}) - Hx]^T + \cdots\} \\
 &= \frac{1}{2(n+\kappa)} \{H \sqrt{1} \sqrt{1}^T H^T + H \sqrt{2} \sqrt{2}^T H^T + \cdots\} \\
 &\quad + \frac{1}{2(n+\kappa)} \{H \sqrt{1} \sqrt{1}^T H^T + H \sqrt{2} \sqrt{2}^T H^T + \cdots\} \\
 &= \frac{2}{2(n+\kappa)} H \{\sqrt{1} \sqrt{1}^T + \sqrt{2} \sqrt{2}^T + \cdots + \sqrt{n} \sqrt{n}^T\} H^T \\
 &= HPH^T
 \end{aligned}$$

where the symbol  $\sqrt{i}$  represents the scaled  $i$  th column/row vector of the covariance matrix factor.

- 8.11** Establish the following computational comparisons between the UKF algorithm and elements of the linear Kalman filter for a linear system using the preceding correspondence

	Number of operations [math operation]				Number of operations [math operation]		
Kalman filter	*	+	Unscented/ Kalman filter	*	+		
Covariance time update $\Phi P \Phi^T$	$2n^3$	$n^2(n - 1)$	Predicted covariance $\sum_{i=0}^{2n} W_i (\tilde{\chi}_i - \bar{x})$ $\times (\tilde{\chi}_i - \bar{x})^{T^T}$		$2(2n + 1)n^2$	$2n(n^2 + 1)$	
Residual variance $HPH^T$	$mn$	$m(n - 1)$	Residual variance $\sum_{i=0}^{2n} W_i (Y_i - \bar{y})$ $\times (Y_i - \bar{y})^{T^T}$	$(n + m)$	$(n + m)$	$2(2n + 1)m^2$	$(2n + 1)m^2 + 2m$

This table suggests that, for the terms compared, the UKF algorithm requires more computational operations than the corresponding linear filter terms.

## **Part 2**

# **Applications**



## Strap-Down Inertial Sensor Laboratory Calibration

Inertial sensors, whose models were described in Chapter 6, form the basis of inertial navigation systems. These sensors are mechanized into an orthogonal triad cluster, one each for accelerometers and for gyros. The availability of accelerations and rates from this configuration was assumed for navigation equations developed in Chapter 5. Before the inertial navigation system's operational use, its accelerometers and gyros are initially calibrated in the laboratory (factory) to remove repeatable errors that are the result of manufacturing imperfections and determine the temperature sensitivity of those errors. After laboratory calibration, remaining errors (errors that are difficult to model effectively, additional temperature variations, etc.) should be small.

Presented in this chapter is a technique that parallels that presented by Diesel [19] to calibrate strap-down inertial sensor errors in a laboratory setting. This is the first application of the elements in Part 1 and addresses the following topics: navigation mechanization review, sensor error model, solutions for sensor errors, data-collection rotation sequences, observation equations, processing sequences, and simulated laboratory data calibration.

It is assumed that a three-axis table is available to perform this calibration. A navigation unit, consisting of accelerometers and gyros as just described, is oriented and rotated in a sequence of nine rotations organized in three sets. Data collected from these rotation sequences provide the means to calibrate the instruments' biases, scale-factor, and misalignment errors over the temperature ranges required by operational considerations. Provided as curve-fit polynomials in temperature, the results of the calibration are used to compensate outputs from sensor components to correct for constant errors and temperature variations.

Data collected are referenced to a navigation solution generated from discrete changes in velocity and attitude from accelerometer and gyro  $\Delta v$  and  $\Delta\theta$  outputs. Navigation velocity states are the principal data used by this calibration technique. Referenced in a local-level frame, velocities should be zero because the unit being calibrated is not translating. It is assumed that the rates of change of velocity error are known, allowing the velocity error equations developed in Chapter 5 to be treated as algebraic equations to solve for the inertial sensor errors. The rates of change of velocity error are obtained from a Kalman filter formulated as a “tracking” filter to reconstruct the rates of change from velocity error measurement updates.

## 9.1 Navigation Mechanization Review

This calibration technique uses a local-level navigation reference frame mechanization from which velocity errors are obtained. Simplifications are made because the laboratory setting's nonmaneuvering 1-g environment. The navigation frame used is a geographic north–east–down (*N*–*E*–*D*) frame.

Inertial navigation system error dynamic equations were developed in Chapter 5. A simplified form of these equations, to include only velocity and tilt errors, is approximated by the following:

$$\delta\dot{v}_x = \delta f_x + g\phi_y \quad (9.1)$$

$$\delta\dot{v}_y = \delta f_y - g\phi_x \quad (9.2)$$

$$\delta\dot{v}_z = \delta f_z \quad (9.3)$$

$$\dot{\phi}_x = \Omega_z\phi_y - \Omega_y\phi_z + \varepsilon_x \quad (9.4)$$

$$\dot{\phi}_y = \Omega_x\phi_z - \Omega_z\phi_x + \varepsilon_y \quad (9.5)$$

$$\dot{\phi}_z = \varepsilon_z \quad (9.6)$$

It is assumed that the transport rate terms, are negligible. These equations show that gravity  $g$  and Earth rate components, that is,  $\Omega_z$ , are environmental inputs which can be used to alter the sensor's outputs in order to determine the accelerometer  $\delta f$  and gyro  $\varepsilon$  errors.

Rotation sequences are tailored such that, by positioning the inertial sensor case, individual case-referenced errors contribute to different components of navigation frame velocity errors. Velocity errors and their rates of change are used to solve for case-referenced sensor errors.

## 9.2 Sensor Error Model

Accelerometer and gyro errors to be calibrated are assumed to consist of constants: bias, scale-factor, and misalignment. The following equations define these contributors in the case frame:

$$\begin{bmatrix} \delta f_x^c \\ \delta f_y^c \\ \delta f_z^c \end{bmatrix} = \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} + \begin{bmatrix} aSF_x & aMA_{xy} & aMA_{xz} \\ aMA_{yx} & aSF_y & aMA_{yz} \\ aMA_{zx} & aMA_{zy} & aSF_z \end{bmatrix} \begin{bmatrix} f_x^c \\ f_y^c \\ f_z^c \end{bmatrix} \quad (9.7)$$

$$\begin{bmatrix} \varepsilon_x^c \\ \varepsilon_y^c \\ \varepsilon_z^c \end{bmatrix} = \begin{bmatrix} gB_x \\ gB_y \\ gB_z \end{bmatrix} + \begin{bmatrix} gSF_x & gMA_{xy} & gMA_{xz} \\ gMA_{yx} & gSF_y & gMA_{yz} \\ gMA_{zx} & gMA_{zy} & gSF_z \end{bmatrix} \begin{bmatrix} \omega_x^c \\ \omega_y^c \\ \omega_z^c \end{bmatrix} \quad (9.8)$$

### 9.3 Solutions for Sensor Errors

In the processing to be described later, Eqs. (9.1–9.6) are manipulated algebraically. At the beginning and end of rotation sequences, rates of change in velocity error are used to compute errors in Eqs. (9.7) and (9.8).

The difference between the two horizontal velocity error components' rates of change, rotating from an initial orientation at  $t = 0$  to a final orientation at  $t = T$ , are given as

$$\delta\dot{v}_x(T) - \delta\dot{v}_x(0) = \Delta\delta f_x + g\Delta\phi_y \quad (9.9)$$

$$\delta\dot{v}_y(T) - \delta\dot{v}_y(0) = \Delta\delta f_y - g\Delta\phi_x \quad (9.10)$$

where

$$\begin{aligned} \Delta\delta f &= C_c^R(T) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} + \begin{bmatrix} aSF_x & aMA_{xy} & aMA_{xz} \\ aMA_{yx} & aSF_y & aMA_{yz} \\ aMA_{zx} & aMA_{zy} & aSF_z \end{bmatrix} \begin{bmatrix} f_x^c(T) \\ f_y^c(T) \\ f_z^c(T) \end{bmatrix} \right) \\ &\quad - C_c^R(0) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} + \begin{bmatrix} aSF_x & aMA_{xy} & aMA_{xz} \\ aMA_{yx} & aSF_y & aMA_{yz} \\ aMA_{zx} & aMA_{zy} & aSF_z \end{bmatrix} \begin{bmatrix} f_x^c(0) \\ f_y^c(0) \\ f_z^c(0) \end{bmatrix} \right) \end{aligned} \quad (9.11)$$

and

$$\Delta\boldsymbol{\phi} = \int_0^T C_c^R(t) \begin{bmatrix} gSF_x & gMA_{xy} & gMA_{xz} \\ gMA_{yx} & gSF_y & gMA_{yz} \\ gMA_{zx} & gMA_{zy} & gSF_z \end{bmatrix} \boldsymbol{\omega}_{\text{sensed}}^T dt \quad (9.12)$$

In Eq. (9.12), the contribution from gyro bias has been temporarily ignored.

Higher derivatives of Eqs. (9.1) and (9.2) will be used to determine some errors. These higher derivatives are approximated as

$$\ddot{\delta v}_x = g\dot{\phi}_y = g(\Omega_x\phi_z - \Omega_z\phi_x + \varepsilon_y) \quad (9.13)$$

$$\ddot{\delta v}_y = -g\dot{\phi}_x = -g(\Omega_z\phi_y - \Omega_y\phi_z + \varepsilon_x) \quad (9.14)$$

where it is assumed that the accelerometer errors  $\delta f$  are constant over the data-collection interval.

### 9.4 Data-Collection Rotation Sequences

Rotation sequences force individual sensor errors to contribute into different components of velocity error and its rate of change. These rotation sequences are grouped into three sets. For each of these three sets, the initial orientation

	$C_c^R(0)$	$f^c(0)$
<b>Set I</b>	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$ $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}$ $\begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$ $\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}$
<b>Set II</b>	$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$ $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ -1 & 0 & 0 \end{bmatrix}$ $\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -g \\ 0 \\ 0 \end{bmatrix}$ $\begin{bmatrix} g \\ 0 \\ 0 \end{bmatrix}$ $\begin{bmatrix} -g \\ 0 \\ 0 \end{bmatrix}$
<b>Set III</b>	$\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ $\begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}$ $\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix}$ $\begin{bmatrix} 0 \\ g \\ 0 \end{bmatrix}$ $\begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix}$

**Fig. 9.1** Rotation sets.

of the sensor case is along a cardinal heading, that is, north. This initial orientation is changed for subsequent rotation sequence sets. Shown in Fig. 9.1 are the initial orientations of the sensor case with respect to these headings. Also shown are the  $\pi$  rotations used for each sequence within the rotation set, starting from its initial orientation.

## 9.5 Observation Equations

Data observed at the beginning and end of the  $\pi$  rotations are used to extract component errors. Data set I, shown in Fig. 9.1, will be used to illustrate the method to solve for errors.

### 9.5.1 Rotation 1

$$\begin{aligned}\Delta\delta f &= C_c^R(T) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} + g \begin{bmatrix} aMA_{xz} \\ aMA_{yz} \\ aSF_z \end{bmatrix} \right) - C_c^R(0) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} - g \begin{bmatrix} aMA_{xz} \\ aMA_{yz} \\ aSF_z \end{bmatrix} \right) \\ &= \begin{bmatrix} 2g & aMA_{xz} \\ -2aB_y & \\ -2aB_z & \end{bmatrix} \quad (9.15)\end{aligned}$$

and

$$\begin{aligned}\Delta\phi &= \int_0^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} gSF_x & - & - \\ gMA_{yx} & - & - \\ gMA_{zx} & - & - \end{bmatrix} \begin{bmatrix} \omega_x \\ 0 \\ 0 \end{bmatrix} dt \\ &= \int_0^\pi \begin{bmatrix} gSF_x \\ gMA_{yx} \cos\varphi - gMA_{zx} \sin\varphi \\ gMA_{yx} \sin\varphi + gMA_{zx} \cos\varphi \end{bmatrix} d\varphi \\ &= \begin{bmatrix} \pi g SF_x \\ -2g MA_{zx} \\ 2g MA_{yx} \end{bmatrix} \quad (9.16)\end{aligned}$$

Substituting these results into Eq. (9.9) yields

$$\begin{aligned}\delta\dot{v}_x(T) - \delta\dot{v}_x(0) &= \Delta\delta f_x + g\Delta\phi_y \\ &= 2(aMA_{xz} - gMA_{zx})g \quad \{\equiv a_1^I\} \quad (9.17)\end{aligned}$$

Similarly, for Eq. (9.10)

$$\begin{aligned}\delta\dot{v}_y(T) - \delta\dot{v}_y(0) &= \Delta\delta f_y - g\Delta\phi_x \\ &= -2aB_y - \pi g gSF_x \quad \{\equiv b_1^I\} \quad (9.18)\end{aligned}$$

### 9.5.2 Rotation 2

$$\begin{aligned}\Delta\delta f &= C_c^R(T) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} - g \begin{bmatrix} aMA_{xz} \\ aMA_{yz} \\ aSF_z \end{bmatrix} \right) - C_c^R(0) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} + g \begin{bmatrix} aMA_{xz} \\ aMA_{yz} \\ aSF_z \end{bmatrix} \right) \\ &= \begin{bmatrix} -2g & aMA_{xz} \\ 2aB_y & \\ 2aB_z & \end{bmatrix} \quad (9.19)\end{aligned}$$

and

$$\begin{aligned}\Delta\phi &= \int_0^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & -\cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & -\cos\varphi \end{bmatrix} \begin{bmatrix} gSF_x & - & - \\ gMA_{yx} & - & - \\ gMA_{zx} & - & - \end{bmatrix} \begin{bmatrix} \omega_x \\ 0 \\ 0 \end{bmatrix} dt \\ &= \int_0^\pi \begin{bmatrix} gSF_x \\ -gMA_{yx}\cos\varphi + gMA_{zx}\sin\varphi \\ -gMA_{yx}\sin\varphi - gMA_{zx}\cos\varphi \end{bmatrix} d\varphi \\ &= \begin{bmatrix} \pi gSF_x \\ 2gMA_{zx} \\ -2gMA_{yx} \end{bmatrix} \quad (9.20)\end{aligned}$$

Substituting these results into Eq. (9.9) yields

$$\begin{aligned}\delta\dot{v}_x(T) - \delta\dot{v}_x(0) &= \Delta\delta f_x + g\Delta\phi_y \\ &= -2(aMA_{xz} - gMA_{zx})g \quad \{\equiv a_2^I\} \quad (9.21)\end{aligned}$$

Similarly, for Eq. (9.10)

$$\begin{aligned}\delta\dot{v}_y(T) - \delta\dot{v}_y(0) &= \Delta\delta f_y - g\Delta\phi_x \\ &= 2aB_y - \pi g gSF_x \quad \{\equiv b_2^I\} \quad (9.22)\end{aligned}$$

### 9.5.3 Rotation 3

$$\begin{aligned}\Delta\delta f &= C_c^R(T) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} - g \begin{bmatrix} aMA_{xz} \\ aMA_{yz} \\ aSF_z \end{bmatrix} \right) - C_c^R(0) \left( \begin{bmatrix} aB_x \\ aB_y \\ aB_z \end{bmatrix} - g \begin{bmatrix} aMA_{xz} \\ aMA_{yz} \\ aSF_z \end{bmatrix} \right) \\ &= \begin{bmatrix} -2aB_x + 2g aMA_{xz} \\ -2aB_y + 2g aMA_{yz} \\ 0 \end{bmatrix} \quad (9.23)\end{aligned}$$

and

$$\begin{aligned}\Delta\phi &= \int_0^T \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} - & - & gMA_{xz} \\ - & - & gMA_{yz} \\ - & - & gSF_z \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \omega_z \end{bmatrix} dt \\ &= \int_0^\pi \begin{bmatrix} gMA_{xz} \cos\psi - gMA_{yz} \sin\psi \\ gMA_{xz} \sin\psi + gMA_{yz} \cos\psi \\ gSF_z \end{bmatrix} d\psi \\ &= \begin{bmatrix} -2gMA_{yz} \\ 2gMA_{xz} \\ \pi gSF_z \end{bmatrix} \quad (9.24)\end{aligned}$$

Substituting these results into Eq. (9.9) yields

$$\begin{aligned}\delta\dot{v}_x(T) - \delta\dot{v}_x(0) &= \Delta\delta f_x + g\Delta\phi_y \\ &= -2aB_x + 2(aMA_{xz} + gMA_{xz})g \quad \{\equiv a_3^I\} \quad (9.25)\end{aligned}$$

Similarly, for Eq. (9.10)

$$\begin{aligned}\delta\dot{v}_y(T) - \delta\dot{v}_y(0) &= \Delta\delta f_y - g\Delta\phi_x \\ &= -2aB_y + 2(aMA_{yz} + gMA_{yz})g \quad \{\equiv b_3^I\} \quad (9.26)\end{aligned}$$

Continuing with rotation sequence 3, Eqs. (9.13) and (9.14) are applied to data at the beginning and end of this rotation. This additional computation permits the tilt errors  $\phi_x$  and  $\phi_y$ , and the azimuth error  $\phi_z$ , to be expressed in terms of the

velocity error rate of change as

$$\begin{aligned}\delta\ddot{v}_x(T) - \delta\ddot{v}_x(0) &= g(-gB_y + \Omega_x\phi_z - \Omega_z\phi_x) - g(gB_y + \Omega_x\phi_z - \Omega_z\phi_x) \\ &= -2g gB_y\end{aligned}\quad (9.27)$$

$$\begin{aligned}\delta\ddot{v}_y(T) - \delta\ddot{v}_y(0) &= -g(-gB_x + \Omega_z\phi_y - \Omega_y\phi_z) + g(gB_x + \Omega_z\phi_y - \Omega_y\phi_z) \\ &= -2g gB_x\end{aligned}\quad (9.28)$$

and

$$\begin{aligned}\delta\ddot{v}_x(T) + \delta\ddot{v}_x(0) &= g(-gB_y + \Omega_x\phi_z - \Omega_z\phi_x) + g(gB_y + \Omega_x\phi_z - \Omega_z\phi_x) \\ &= -2g \Omega_z\phi_x + 2g \Omega_x\phi_z\end{aligned}\quad (9.29)$$

Returning to Eqs. (9.1) and (9.2), the following are obtained:

$$\begin{aligned}\delta\dot{v}_x(T) + \delta\dot{v}_x(0) &= -\delta f_x + g\phi_y + (\delta f_x + g\phi_y) \\ &= 2g \phi_y\end{aligned}\quad (9.30)$$

$$\begin{aligned}\delta\dot{v}_y(T) + \delta\dot{v}_y(0) &= -\delta f_y - g\phi_x + (\delta f_y - g\phi_x) \\ &= -2g \phi_x\end{aligned}\quad (9.31)$$

Finally, the following result is obtained from Eq. (9.6):

$$\dot{\phi}_z(T) - \dot{\phi}_z(0) = -2gB_z \quad (9.32)$$

where the assumption of the availability of the azimuth misalignment rate of change is made.

A short-hand convention is used to summarize the preceding results and for the other rotation sets shown in Fig. 9.1 [17]. This summary for the observations is presented in Table 9.1. The short-hand convention is

$$\begin{aligned}\text{rotation } i, \text{ set } j; \quad a_i^j &\equiv \delta\dot{v}_x(T) - \delta\dot{v}_x(0) \\ b_i^j &\equiv \delta\dot{v}_y(T) - \delta\dot{v}_y(0)\end{aligned}\quad (9.33)$$

Errors for set II and set III can be obtained from those of set I by permuting the subscripts. Solutions for these errors are summarized in Table 9.2.

## 9.6 Processing Sequences

Data can be collected rapidly for each of the rotation sets and sequences within each set. As little as 5–10 s at each observation point is sufficient [17]. The nine orientation points can be accomplished in a few minutes. At the completion of rotation 3 of the first set, tilt and azimuth errors are computed. These are

Table 9.1 Observation equations for errors

Rotation	Set I		Set II		Set III	
	$d_i^I$	$b_i^I$	$d_i^{II}$	$b_i^{II}$	$d_i^{III}$	$b_i^{III}$
Rotation 1	$d_1^I = 2(aMA_{xz} - gMA_{xy})g$	$b_1^I = -2aB_y - \pi g SF_x$	$d_1^{II} = 2(aMA_{zy} - gMA_{yz})g$	$b_1^{II} = -2aB_z - \pi g SF_y$	$d_1^{III} = 2(aMA_{zy} - gMA_{yz})g$	$b_1^{III} = -2aB_x - \pi g SF_z$
	$d_2^I = -2(aMA_{xz} - gMA_{xy})g$	$b_2^I = 2aB_y - \pi g SF_x$	$d_2^{II} = -2(aMA_{yx} - gMA_{xy})g$	$b_2^{II} = 2aB_z - \pi g SF_y$	$d_2^{III} = -2(aMA_{zy} - gMA_{yz})g$	$b_2^{III} = 2aB_x - \pi g SF_z$
	$d_3^I = -2(aMA_{xz} + 2(aMA_{xy} + gMA_{yz})g$	$b_3^I = -2aB_y + 2(aMA_{xy} + gMA_{yz})g$	$d_3^{II} = -2aB_y + 2(aMA_{yx} + gMA_{xy})g$	$b_3^{II} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$d_3^{III} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$b_3^{III} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$
	$d_4^I = -2aB_y + 2(aMA_{yz} + gMA_{xy})g$	$b_4^I = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$d_4^{II} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$b_4^{II} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$	$d_4^{III} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$	$b_4^{III} = -2aB_y + 2(aMA_{yz} + gMA_{yz})g$
Rotation 2	$d_1^I = -2(aMA_{xz} - gMA_{xy})g$	$b_1^I = -2aB_z - \pi g SF_y$	$d_1^{II} = -2(aMA_{zy} - gMA_{yz})g$	$b_1^{II} = -2aB_x - \pi g SF_z$	$d_1^{III} = -2(aMA_{zy} - gMA_{yz})g$	$b_1^{III} = -2aB_z - \pi g SF_z$
	$d_2^I = -2(aMA_{xz} - gMA_{xy})g$	$b_2^I = 2aB_z - \pi g SF_y$	$d_2^{II} = -2(aMA_{yx} - gMA_{xy})g$	$b_2^{II} = 2aB_x - \pi g SF_z$	$d_2^{III} = -2(aMA_{zy} - gMA_{yz})g$	$b_2^{III} = 2aB_x - \pi g SF_z$
	$d_3^I = -2(aMA_{xz} + 2(aMA_{xy} + gMA_{yz})g$	$b_3^I = -2aB_z + 2(aMA_{xy} + gMA_{yz})g$	$d_3^{II} = -2aB_y + 2(aMA_{yx} + gMA_{xy})g$	$b_3^{II} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$	$d_3^{III} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$b_3^{III} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$
	$d_4^I = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$b_4^I = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$	$d_4^{II} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$	$b_4^{II} = -2aB_y + 2(aMA_{yz} + gMA_{yz})g$	$d_4^{III} = -2aB_y + 2(aMA_{yz} + gMA_{yz})g$	$b_4^{III} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$
Rotation 3	$d_1^I = -2(aMA_{xz} - gMA_{xy})g$	$b_1^I = -2aB_y - \pi g SF_x$	$d_1^{II} = -2(aMA_{zy} - gMA_{yz})g$	$b_1^{II} = -2aB_z - \pi g SF_y$	$d_1^{III} = -2(aMA_{zy} - gMA_{yz})g$	$b_1^{III} = -2aB_x - \pi g SF_z$
	$d_2^I = -2(aMA_{xz} - gMA_{xy})g$	$b_2^I = 2aB_y - \pi g SF_x$	$d_2^{II} = -2(aMA_{yx} - gMA_{xy})g$	$b_2^{II} = 2aB_z - \pi g SF_y$	$d_2^{III} = -2(aMA_{zy} - gMA_{yz})g$	$b_2^{III} = 2aB_x - \pi g SF_z$
	$d_3^I = -2(aMA_{xz} + 2(aMA_{xy} + gMA_{yz})g$	$b_3^I = -2aB_y + 2(aMA_{xy} + gMA_{yz})g$	$d_3^{II} = -2aB_y + 2(aMA_{yx} + gMA_{xy})g$	$b_3^{II} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$d_3^{III} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$b_3^{III} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$
	$d_4^I = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$	$b_4^I = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$	$d_4^{II} = -2aB_x + 2(aMA_{xy} + gMA_{xy})g$	$b_4^{II} = -2aB_y + 2(aMA_{yz} + gMA_{yz})g$	$d_4^{III} = -2aB_y + 2(aMA_{yz} + gMA_{yz})g$	$b_4^{III} = -2aB_z + 2(aMA_{xy} + gMA_{xy})g$

Table 9.2 Solution for errors

	Set I	Set II	Set III
Gyro scale-factor	$gSF_x = \frac{b'_2 + b'_1}{-2\pi g}$	$gSF_y = \frac{b''_2 + b''_1}{-2\pi g}$	$gSF_z = \frac{b'''_2 + b'''_1}{-2\pi g}$
Accelerometer bias	$aB_y = \frac{b'_2 - b'_1}{4}$	$aB_z = \frac{b''_2 - b''_1}{4}$	$aB_x = \frac{b'''_2 - b'''_1}{4}$
Axis definition	$aMA_{xz} \equiv 0$ and $aMA_{yz} \equiv 0$	$aMA_{yx} \equiv 0$	—
Gyro misalignment	$gMA_{yz} = \frac{b'_3 + 2aB_y}{2g}$	$gMA_{yx} = \frac{a''_3 + 2aB_z}{2g}$	$gMA_{xy} = \frac{a'''_3 + 2aB_z}{2g} - aMA_{zy}$
Accelerometer misalignment	$gMA_{zx} = \frac{a'_3 + 2aB_x}{2g}$	$aMA_{zx} = \frac{b''_3 + 2aB_z}{2g} - gMA_{xy}$	$aMA_{xy} = \frac{b'''_3 + 2aB_x}{2g} - gMA_{yz}$
	—	—	$aMA_{zy} = \frac{a'''_3 + 2aB_x}{2g} + gMA_{zx}$

incorporated as compensations for subsequent computations. This is consistent with an assumption necessary to obtain Eqs. (9.13) and (9.14).

Sensor error trends with temperature can be obtained from several repeated cycles through error computations. During each cycle, computed errors are recorded, as well as the measured temperature (or average) during that cycle (starting at a sufficiently low temperature, allowing the temperature to increase during testing, and continuing to a sufficiently high temperature). Recorded data are then processed using linear regression analysis or least-squares curve fit, as presented in Example 2.3, to obtain a best-fit model for the error coefficients as a function of temperature.

## 9.7 Simulated Laboratory Data Calibration

To illustrate the use of the error equations developed, a simulated inertial sensor's data are generated for the rotation set I. The simulation's navigation equations are implemented in a local-level *N-E-D* frame. The inertial navigation sensor errors included for this illustration are accelerometer and gyro biases and scale-factor errors. Accelerometer and gyro errors simulated assume low-quality inertial sensors. Each sensor's error is based on a random draw with a variance that corresponds to the sensor's expected  $1-\sigma$  error.

The process assumes the availability of rate of change of velocity error. This availability is satisfied by using a tracking model, implemented in a Kalman filter algorithm, to produce estimates of the rate of change of velocity error. The tracking filter dynamics model for the  $i$ th velocity component is

$$\frac{d}{dt} \begin{bmatrix} \delta v_i \\ \delta \dot{v}_i \\ \delta \ddot{v}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta v_i \\ \delta \dot{v}_i \\ \delta \ddot{v}_i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ w_i \end{bmatrix} \quad (9.34)$$

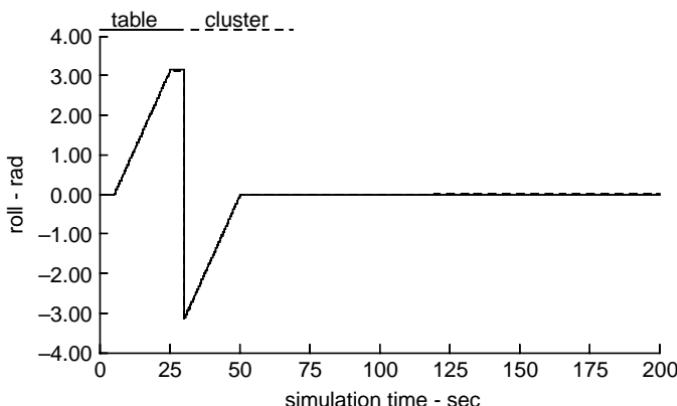


Fig. 9.2 Laboratory-simulated inertial cluster roll.

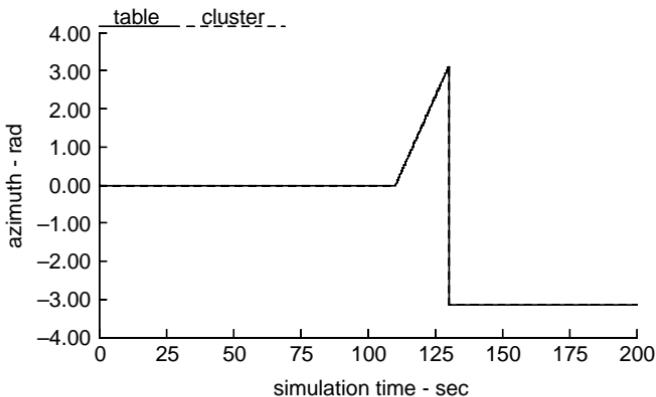


Fig. 9.3 Laboratory-simulated inertial cluster azimuth.

The corresponding measurement matrix is

$$\mathbf{h}^T = [1 \ 0 \ 0] \quad (9.35)$$

Each velocity component is processed in its unique filter algorithm

$$z_i = v_i - \delta v_i \quad (9.36)$$

Figures 9.2 and 9.3 present typical simulated attitude time histories for the rotation set I. Figures 9.4 and 9.5 show typical simulated north and east navigation frame velocity and estimated velocity errors. In Figs. 9.6 and 9.7 are the corresponding estimated rate of change for those velocity components. Included in these plots is the tracking filter's  $1-\sigma$  estimate uncertainty. These later figures show that the filter's estimate of rate of change of velocity error contains

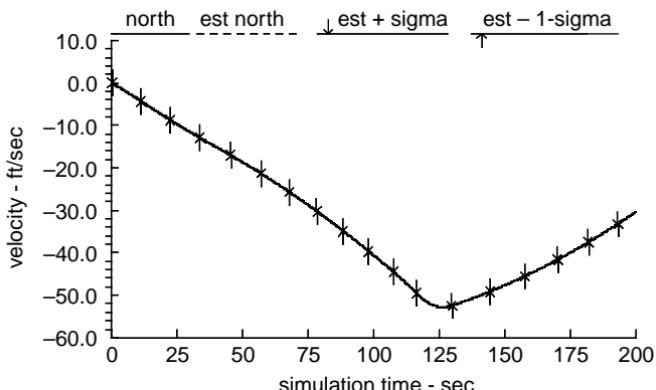
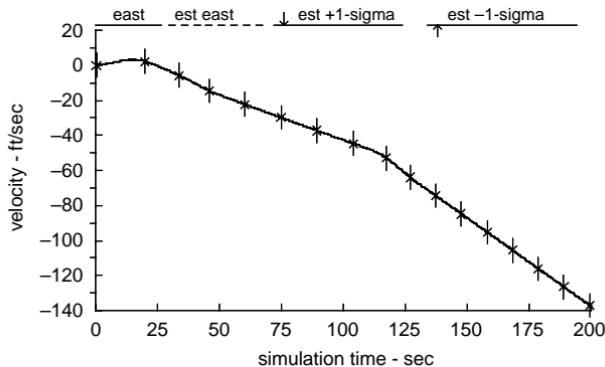
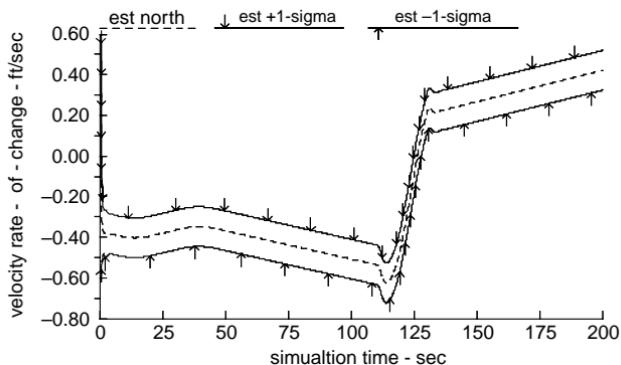


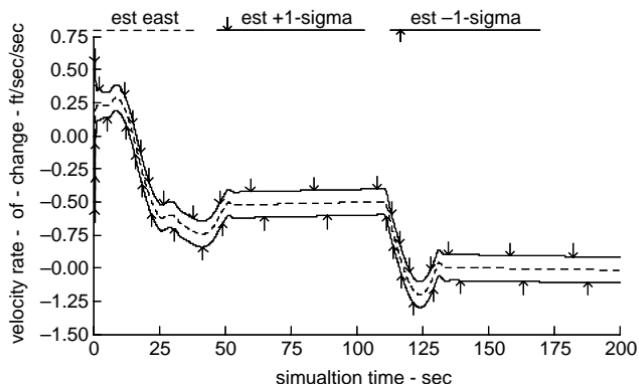
Fig. 9.4 Simulated inertial cluster north velocity and filter's estimate.



**Fig. 9.5 Simulated inertial cluster east velocity and filter's estimate.**



**Fig. 9.6 Filter's estimate of inertial cluster north velocity rate of change.**



**Fig. 9.7 Filter's estimate of inertial cluster east velocity rate of change.**

**Table 9.3** Simulation results for rotation set I<sup>a</sup>

Error	True	Computed
Tilt, rad		
$\phi_x$	0.02334	0.02328
$\phi_y$	-0.00528	-0.00498
$\phi_z$	—	—
Accelerometer bias, ft/s <sup>2</sup>		
$aB_x$	—	—
$aB_y$	0.2298	0.2294
$aB_z$	—	—
Gyro bias, rad/s		
$gB_x$	0.1443E-4	0.1782E-4
$gB_y$	0.8941E-4	0.9054E-4
$gB_z$	—	—
Gyro scale-factor		
$gSF_x$	-0.00381	-0.00393
$gSF_y$	—	—
$gSF_z$	—	—

<sup>a</sup>Note: With just the first rotation set, not all of the instrument errors can be computed (see Table 9.2).

additional dynamics (transient oscillations). Also, the amount of instrument error governs the accuracy of filter tracking and corresponding estimates.

Filter estimates required are captured at instants before and after the rotations. From these estimates, attitude (tilt) and instrument errors are computed. Table 9.3 summarizes the results for this simulation illustration.

## 9.8 Chapter Summary

Presented in this chapter was a technique that parallels that presented by Diesel [17] to calibrate strap-down inertial sensor errors in a laboratory setting. Velocity error equations developed in Chapter 5 are treated as algebraic equations, assuming that the rates of change of velocity error are known, to solve for inertial sensor errors. These rates of change of velocity error are obtained from a Kalman filter formulated as a tracking filter to reconstruct the rates of change from velocity error measurement updates. Sensor error equations are developed and used to compute inertial sensor errors from simulated laboratory data.

The Kalman filter implemented in this case study is not representative of an implementation used in an integrated navigation system, but was a filter tailored to generate higher derivatives, through its state modeling and corresponding estimates.

Laboratory calibration provides an estimate of stable sensor errors; however, the process is imperfect, and additional sensor error remains even after calibration. The need for refined estimates of inertial sensors errors in a postlaboratory calibration can be accomplished by including remainder sensor errors in Kalman filters used during ground alignment (see Sect 11.2) and aided operations (see Chapters 12 and 13).

## 10

# Flight-Test Evaluations

This second application of the elements in Part I addresses navigation systems' flight-test evaluations. Laboratory calibrations presented in the last chapter attempt to reduce inertial-navigation systems' instrument errors. The accuracy of calibrations governs the performance of the inertial-navigation system. In a laboratory environment, attempts are made to duplicate the operational dynamic environment; however, the true test of the system's performance must be assessed while in the actual operational environment.

Dynamic tests are conducted to replicate the inertial-navigation system's operational environment, with additional instrumentation to evaluate the results observed during testing. Unlike the laboratory, the navigation system is moving, and its errors and performance characteristics are more difficult to determine because the exact knowledge of the navigation states, position, velocity, etc., is not known.

Errors in the navigation system are determined, and its performance is characterized by comparing navigation states to "truth" data. For example, if known, the navigation system's indicated position while near a landmark can be compared to the landmark's known position to determine position error. The precision to which the landmark's position is known and, more importantly, the ability to determine the navigation system's position relative to the landmark must be better than the accuracy level within which it is desired to determine the navigation system's error/performance characteristics.

A variety of techniques has been developed to dynamically determine a navigation system's position to a high degree of accuracy. These include optical and radar tracking, posttest processed differential global positioning system, etc. Postflight processing can estimate the navigation system's errors more accurately than the landmark technique.

In this chapter, the following three postflight trajectory reconstruction techniques will be presented: optical tracking trajectory, tactical-air-navigation (TACAN)/inertial-navigation unit (INU), and vehicle dynamics with radar tracking trajectory.

The first and simplest trajectory reconstruction technique uses optical tracking data, which is a nonlinear function of trajectory coordinates (see Example 3.1), to reconstruct the trajectory. With continuous tracking data available, aircraft positions are determined from a least-squares estimator using linearized line-of-sight measurements from two cameras to predict future aircraft positions. This technique is limited in that only the aircraft's position states are determined, and

there is no additional modeling to allow a smoother transition from one position to the next.

The second technique, based on TACAN ranging, is used to reconstruct the navigation states of position, velocity, and attitude. This technique combines data from the INU under test with TACAN ranging in a Kalman filter to obtain corrected navigation states: position, velocity, etc. The objective of flight evaluations, which lead to the development of this TACAN-aided trajectory reconstruction technique, is to determine the position error time history of an integrated Doppler/inertial integrated navigation system. The combination of TACAN with INU reference state data provides a means to smoothly transition from one measurement time instant to the next based on the INU's navigation state data. The integration of INU and TACAN data, via a Kalman filter, overcomes the erratic nature of raw TACAN data, as will be seen in the results presented. Results indicate a reconstructed position repeatability of a few hundred feet.

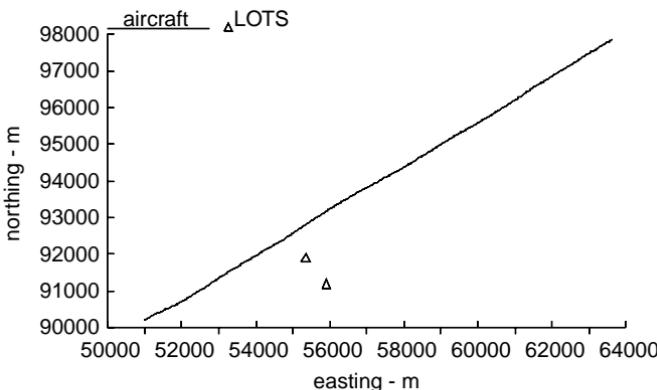
The final technique, a trajectory reconstruction for a ground-launched sounding rocket, is presented. Although not a navigation system, this application involves the same elements as an integrated navigation system: a reference dynamic model, its linearization, and observations of the vehicle's motion. Using vehicle translational equations of motion, including applied aerodynamic and gravitational forces, a Kalman filter algorithm is used to reconstruct the vehicle's system states, position and velocity, based on ground-based radar observations. The primary objective of this application is to estimate payload (ram air inflatable ballute) velocity as it is descending through lower levels of the atmosphere. Intermittent radar tracking data necessitate the use of a dynamic model to smoothly transition from one measurement time instant to the next. Results show a smooth estimate of east velocity that likely corresponds to that component of wind.

The objective in presenting these examples is to illustrate different approaches for independently estimating a vehicle's trajectory and other states as applicable. Results presented do not represent best possible reconstructions but are shown to establish insights into characteristics of estimation algorithms used.

## 10.1 Optical Tracking Trajectory Reconstruction

A lightweight optical tracking system (LOTS) is used as flight-test instrumentation. Light-weight optical tracking system cameras are located at surveyed sites. The placement of two of these cameras in the universal transverse mercator (UTM) grid is illustrated in Figs. 10.1 and 10.2. Also shown are trajectory data produced by Fort Bragg, North Carolina, from LOTS camera data. For this example, we will assume that the UTM conforms to a Cartesian coordinate system, rather than the projection that actually defines the UTM system.

In Example 3.1, equations defining ground-based azimuth and elevation measurements and the variations of these angles with respect to Cartesian coordinates were presented. Equations from this example are used in a least-squares recursive algorithm to compute the position of the aircraft. Measurements are



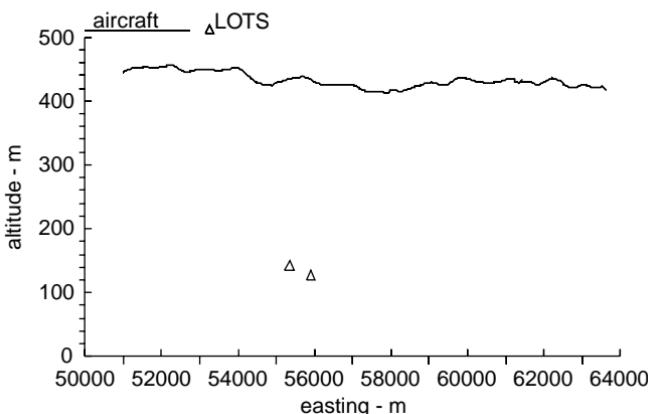
**Fig. 10.1 LOTS camera and aircraft northing and easting positions.**

nonlinear functions of the relative position between the aircraft and LOTS cameras. This measurement can be expressed in general form by momentarily neglecting additive measurement error

$$\mathbf{z} = \mathbf{h}(\mathbf{p}, t) \quad (10.1)$$

Consider the following Taylor-series expansion in vector form relating the anticipated measurement  $\mathbf{z}_{i+1}$  caused by the change in aircraft position from the  $i$ th time instant to the  $i + 1$ th time instant:  $\mathbf{p}_i$  to  $\mathbf{p}_{i+1}$

$$\mathbf{z}_{i+1} = \mathbf{h}(\mathbf{p}_i, t) + \frac{\partial \mathbf{h}(\mathbf{p}_i, t)}{\partial \mathbf{p}} (\mathbf{p}_{i+1} - \mathbf{p}_i) + HOT \quad (10.2)$$



**Fig. 10.2 LOTS camera and aircraft altitudes.**

Or rearranging and including additive errors as shown:

$$z_{i+1} - \mathbf{h}(\mathbf{p}_i, t) = \frac{\partial \mathbf{h}(\mathbf{p}_i, t)}{\partial \mathbf{p}} (\mathbf{p}_{i+1} - \mathbf{p}_i) + \text{errors} \quad (10.3)$$

This form is similar to the measurements equation described in Example 2.3.

The recursive least-squares estimation algorithm is then given by

$$\mathbf{p}_{i+1} = \mathbf{p}_i + G_i [z_{i+1} - \mathbf{h}(\mathbf{p}_i, t)] \quad (10.4)$$

where

$\mathbf{p}_i$  =  $i$ th time computed position

$z_i$  = actual measurement, that is, azimuth from camera 1

$\mathbf{h}(\mathbf{p}_i, t)$  = computed measurement based on prior computed positions

and

$$G_i = (H_i^T H_i)^{-1} H_i^T \quad (10.5)$$

where

$$H_i = \frac{\partial \mathbf{h}(\mathbf{p}_i, t)}{\partial \mathbf{p}} \quad (10.6)$$

The partial derivative matrix in Eq. (10.6) results in an overdetermined problem, that is, two sets of two angle measurements from two LOTS trackers, yielding more rows than columns. This recursive algorithm is similar to a Kalman filter; however, it can only compute an updated position based on the difference between actual and computed measurements.

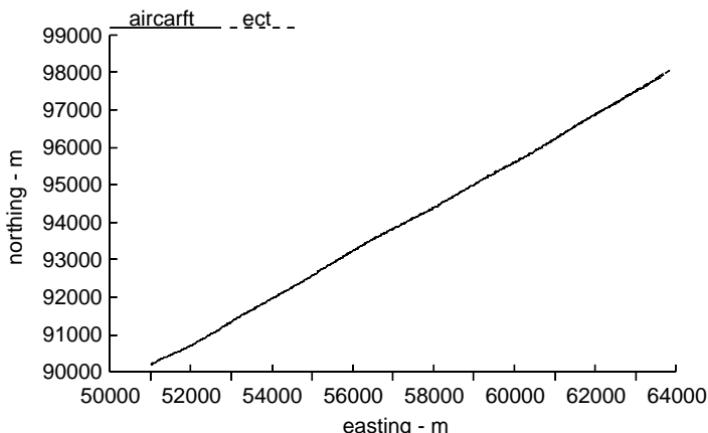
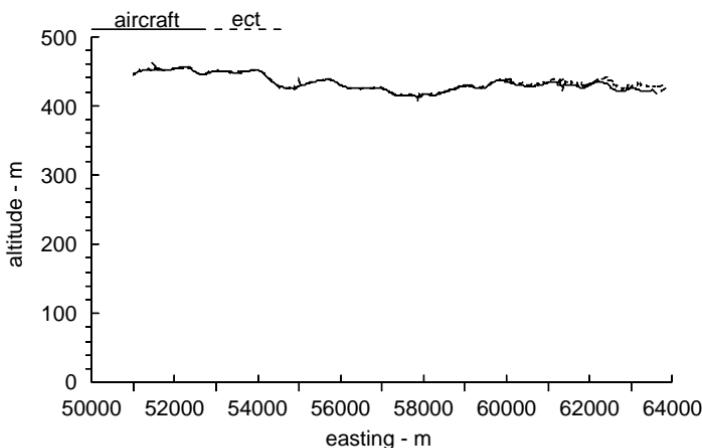


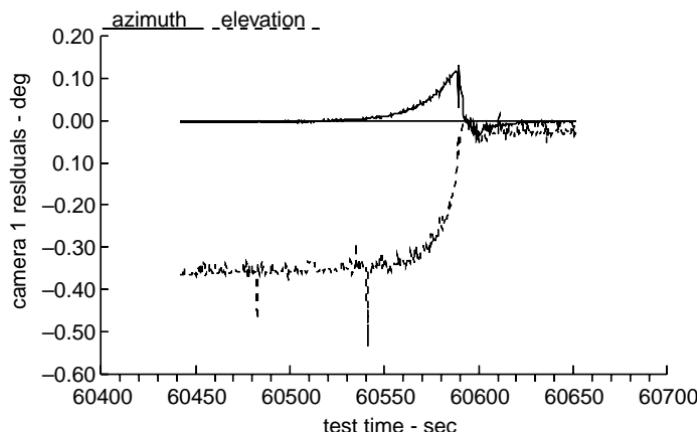
Fig. 10.3 Reconstructed aircraft northing and easting positions.



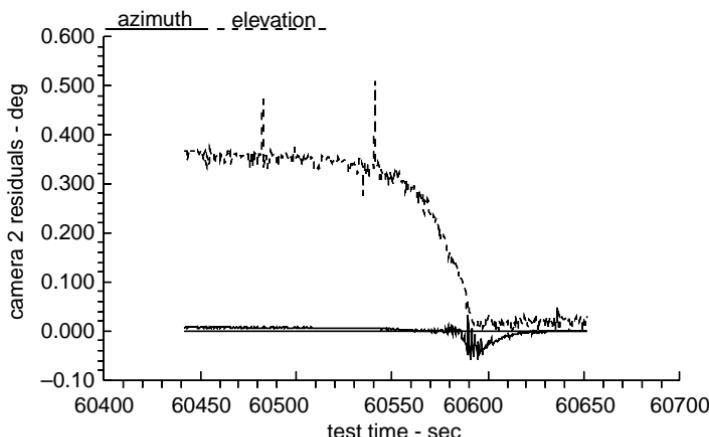
**Fig. 10.4 Reconstructed aircraft altitudes.**

The reconstructed aircraft position resulting from the application of this algorithm is shown in Figs. 10.3 and 10.4. Also shown are data provided by Fort Bragg as the “aircraft” position. The aircraft’s initial position is to the northeast and proceeds to the southwest. There is a difference between these results and those provided by Fort Bragg at extended distances from LOTS positions. As the relative positions become closer, these differences diminish.

To further examine the results of this reconstruction, measurement residuals, azimuth, and elevation for the two LOTS cameras are presented in Figs. 10.5 and 10.6. Residuals reflect the quality of the estimates from this nonlinear estimator. These residuals show that elevation residuals are much larger than azimuth residuals and that the two elevation and two azimuth sets are almost mirror



**Fig. 10.5 LOTS camera 1 measurement residuals.**



**Fig. 10.6 LOTS camera 2 measurement residuals.**

images of the others. It is apparent from the residuals that the LOTS cameras were not tracking the aircraft closely during the entire time segment. Tracking appears to improve at approximately 60,570 s. This corresponds to the time that the aircraft passes the camera positions (see Fig. 10.1) heading southwest. After this time, residuals are much nearer to zero. Correspondingly, the quality of the reconstruction improves, as shown in Figs. 10.3 and 10.4.

## 10.2 Tactical-Air-Navigation/Inertial-Navigation-Unit Reconstruction

### 10.2.1 *Tactical-Air-Navigation/Inertial-Navigation-Unit Kalman Filter States*

The Kalman filter used in this flight evaluation implements a geographic navigation reference frame system error model. Inertial-navigation-unit and TACAN error states included in the filter are as follows:

$\delta r_n$  = north position error, ft

$\delta r_e$  = east position error, ft

$\delta r_d$  = down position error, ft

$\delta v_n$  = north velocity error, ft/s

$\delta v_e$  = east velocity error, ft/s

$\delta v_d$  = down velocity error, ft/s

$\phi_n$  = tilt error about north, rad

$\phi_e$  = tilt error about east, rad

$\phi_d$  = azimuth error, rad

$b_{\text{TACAN}}$  = TACAN range bias error, ft

The preceding error states do not include states associated with accelerometer or gyro errors. Estimation of these errors was not part of the flight-evaluation objectives. Filter process noise in the velocity and tilt error states is used to accommodate this modeling error for missing accelerometer and gyro errors.

$$\begin{bmatrix}
-\frac{v_d}{R} & 0 & \frac{v_n}{R} & 1 & 0 & 0 & 0 & 0 & 0 \\
\frac{v_e t \Phi}{R} & -\left(\frac{v_d}{R} + \frac{v_n}{R} t \Phi\right) & \frac{v_e}{R} & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{v_d}{R} & 0 & 1 & 0 & 0 & 0 \\
\frac{\delta r_n}{\delta v_d} & 0 & \frac{v_n v_d - v_e^2 t \Phi}{R^2} & \frac{v_d}{R} & 2(p + \omega)_d & -\rho_e & 0 & -f_d & f_e \\
\frac{\delta r_e}{\delta v_e} & \frac{v_e v_d + v_n v_e t \Phi}{R^2} & -\left(p + 2\omega\right)_d & -\left(p + 2\omega\right)_d & \frac{v_d + v_n}{R} \Phi & (p + 2\omega)_n & f_e & 0 & -f_n \\
\frac{\delta v_n}{\delta v_d} & 0 & -\frac{(v_n^2 + v_e^2)}{R^2} + 2\frac{s}{R} & 2\rho_e & -2(p + \omega)_n & 0 & -f_e & f_n & 0 \\
\frac{\delta v_e}{\delta v_e} & \frac{v_e^2 2\omega_{i,e} s \Phi}{R} & \frac{v_e}{R^2} & 0 & \frac{1}{R} & 0 & 0 & (\rho + \omega)_n & -\rho_e \\
\frac{d}{dt} \frac{\delta v_e}{\delta v_d} & -\frac{\omega_{i,e} s \Phi}{R} & \frac{v_n}{R^2} & -\frac{v_n}{R^2} & 0 & 0 & -(\rho + \omega)_n & 0 & (\rho + \omega)_n \\
\frac{d}{dt} \frac{\delta v_e}{\delta \phi_n} & 0 & -\frac{v_e t \Phi}{R^2} & 0 & -\frac{t \Phi}{R} & 0 & \rho_e & -(\rho + \omega)_n & 0 \\
\frac{d}{dt} \frac{\delta v_e}{\delta \phi_e} & 0 & -\left(\frac{\omega_{i,e} c \Phi}{R} + \frac{v_e}{R^2 c^2 \Phi}\right) & 0 & 0 & 0 & \rho_e & -(\rho + \omega)_n & 0
\end{bmatrix}
+
\begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
\delta f_n & \delta g_n & 0 \\
\delta f_e + \delta g_e & 0 & 0 \\
\delta f_d + \delta g_d & 0 & 0 \\
0 & \epsilon_n & \epsilon_e \\
0 & \epsilon_e & \epsilon_d \\
0 & 0 & 0
\end{bmatrix}$$

Fig. 10.7 INU system error equation.

The error dynamics model for INU system errors is shown here and is derived from the equations in Chapter 5 for the  $\phi$  angle formulation.

The INU's vertical axis is controlled by a barometric altimeter and is assumed to be decoupled from the horizontal axes. Consequently, the vertical position and velocity error components were assumed to be zero.

### 10.2.2 *Tactical-Air-Navigation Measurements*

Measurement equations for TACAN ranging as a nonlinear function of the geographic positions relative to a TACAN station were presented in Sec. 7.2.

### 10.2.3 *Tactical-Air-Navigation Filter Outputs*

The filter produces estimates of position, velocity, and attitude errors referenced in the geographic frame as its navigation frame. These estimates are combined with the INU's position to produce a corrected position in the Earth-centered Earth-fixed frame using the following equation:

$$\hat{\mathbf{r}}^e = \mathbf{r}_a^e + C_{g_a}^e \delta\hat{\mathbf{r}}^g \quad (10.7)$$

where

$$\delta\mathbf{r}^g = \begin{bmatrix} \delta r_n \\ \delta r_e \\ \delta r_d \end{bmatrix} \quad (10.8)$$

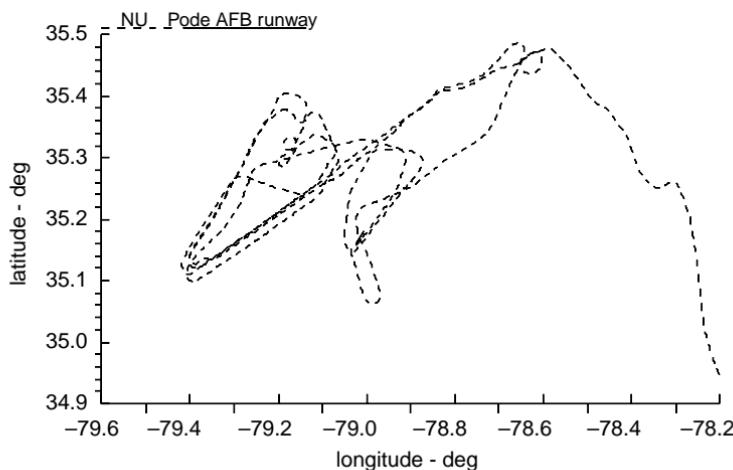
### 10.2.4 *Flight-Test Results*

The Kalman filter just described was used to generate corrected positions for a test flight around Pope Air Force Base, North Carolina. During this flight, the aircraft flew over the runway. Assuming that the aircraft was physically over the runway and comparing reconstructed positions to known runway positions, an additional quality assessment of reconstructed INU states can be made.

Shown in Fig. 10.8 are aircraft position data from the INU (uncorrected). Shown in Fig. 10.9 are corrected position data using position estimates from the postflight Kalman filter. While nearing the TACAN station, which is near the runway, range measurement updates to the filter algorithm are suspended near this “zone of confusion.”

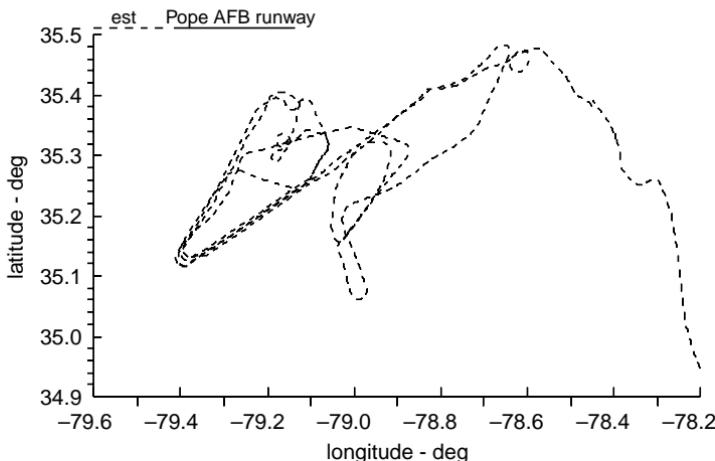
Initially, while the aircraft is flying at a nearly constant radius relative to the TACAN station, there is a delay in the filter’s convergence, as shown for the estimated north and east position errors in Figs. 10.10 and 10.11. Superimposed on these plots are the filter’s  $\pm 1$ -sigma estimation error bounds (square root of error covariance matrix diagonal elements). Estimated INU position errors exhibit 84-min Schuler period characteristic. With the plot origin set at the TACAN station location and the scale expanded, Fig. 10.12 presents reconstructed INU positions during runway flyovers. This figure demonstrates the repeatability of the TACAN/INU filter’s reconstruction at approximately 150 ft.

Another quality measure of the filter’s INU position error estimates just shown is the measurement residuals presented in Fig. 10.13. These residuals appear

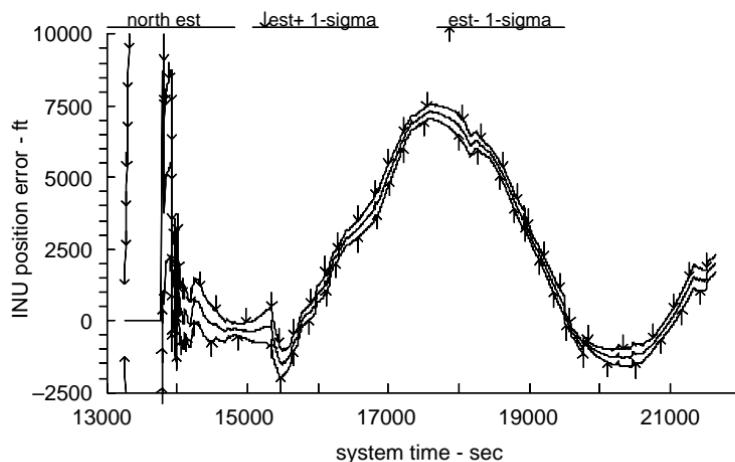


**Fig. 10.8** Flight-test INU positions.

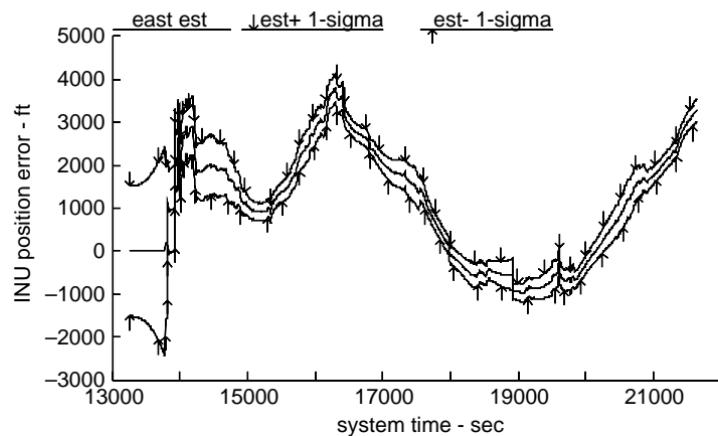
reasonable, with little bias seen, and most of the residuals are lying within the  $1-\sigma$  upper and lower bounds. The variance assumed for the TACAN range error is a tradeoff between the desire for smoothness of estimates and a recognition that the error state structure does not include other errors, that is, INU accelerometer and gyro errors, and that measurements are nonlinearly related to the filter's states. A smaller range variance would result in even more erratic estimates for INU position errors. Several instances of measurement exclusions were required using the residual editing feature described in Chapter 8 (see Problem 8.5). This is indicated when measurement covariances are set artificially to zero.



**Fig. 10.9** Reconstructed trajectory.



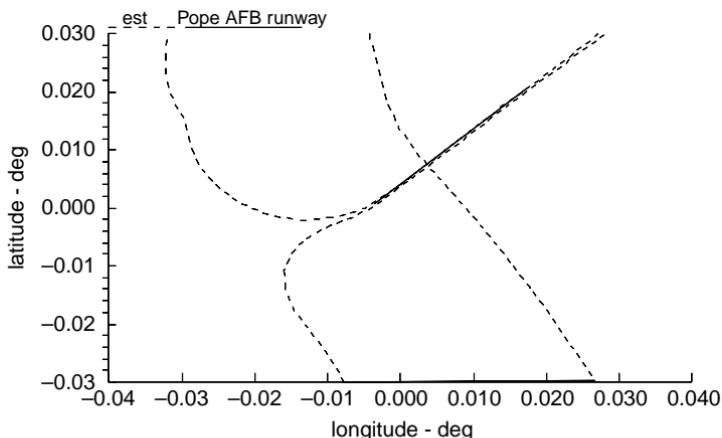
**Fig. 10.10 Estimated north position error.**



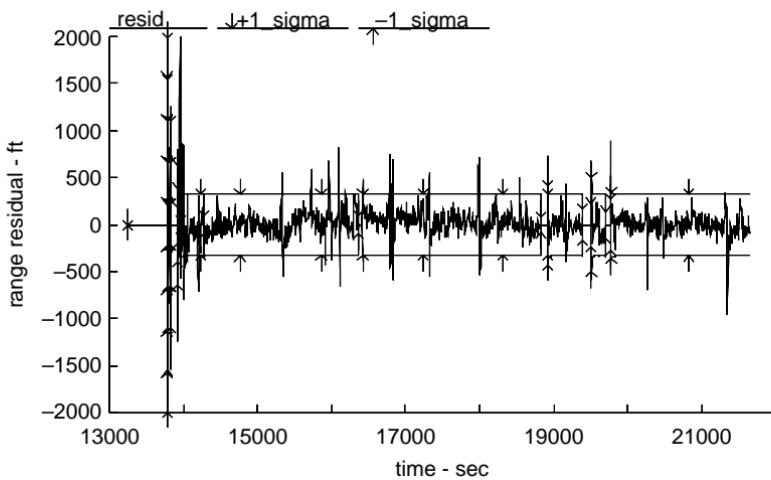
**Fig. 10.11 Estimated east position error.**

### 10.3 Vehicle Dynamics with Radar Tracking Trajectory Reconstruction

The following case study is not a problem in integrated navigation per se, but does involve similar processes—dynamic model linearization and measurements used in Kalman filter to provide optimal estimates of the dynamic states. In this application, gravity and atmospheric drag forces as they affect a vehicle's translational dynamics are used to establish a reference trajectory. This approach differs from the one in Sec. 10.2, which assumes a reference trajectory governed by an inertial-navigation system using sensed accelerations



**Fig. 10.12 Expanded scale reconstructed trajectory referenced to TACAN.**



**Fig. 10.13 TACAN range measurement residual.**

and rates. In this application, the vehicle's translational dynamics, in a central force gravitational field and subject to inertial frame referenced aerodynamic forces, are developed.

### 10.3.1 Position/Velocity Equations

The evolution of position/velocity is obtained from the laws of motion and is given by the following equation:

$$\ddot{\mathbf{r}}^i = \mathbf{a}_D^i + \mathbf{G}^i \quad (10.9)$$

The atmospheric drag is modeled as

$$\begin{aligned}\mathbf{a}_D^i &= -\frac{1}{2} \rho \frac{C_D S_{\text{ref}}}{m} \|\mathbf{v}_{\text{rel}}^i\| \mathbf{v}_{\text{rel}}^i \\ &\equiv -\frac{1}{2} \rho K_D \|\mathbf{v}_{\text{rel}}^i\| \mathbf{v}_{\text{rel}}^i\end{aligned}\quad (10.10)$$

where

$\rho$  = atmospheric density

$C_D$  = drag coefficient

$S_{\text{ref}}$  = aerodynamic drag reference area

$m$  = vehicle mass

$K_D$  = ballistic coefficient (Note:  $S_{\text{ref}}$  and  $m$  are assumed constant.)

This model assumes that the inertial frame is the preferential direction for forces acting on the vehicle and that there is no preferential orientation for these forces; that is,  $K_D$  is not a function of other variables.

The relative velocity in Eq. (10.10) is

$$\mathbf{v}_{\text{rel}}^i \equiv \dot{\mathbf{r}}^i - \mathbf{v}_{\text{wind}}^i \quad (10.11)$$

where the wind velocity in the Earth-centered inertial frame is

$$\mathbf{v}_{\text{wind}}^i \equiv \boldsymbol{\omega}_{i/e}^i \times \mathbf{r}^i \quad (10.12)$$

assuming a rigid atmosphere.

The atmospheric density in Eq. (10.10) is modeled as a simple exponential profile as

$$\rho = \rho_0 e^{-h/H_s} \quad (10.13)$$

where

$\rho_0$  = density at the atmospheric model's reference altitude

$H_s$  = density scale height relative to the model's reference altitude

$h$  = altitude above the model's reference altitude.

### 10.3.2 Position/Velocity Error Equations

The evolution of position and velocity cannot be determined perfectly. Knowledge of the terms on the right-hand side (RHS) of Eq. (10.9) is only approximate. There is inexact knowledge of 1) the vehicle's aerodynamic drag coefficient  $C_D$  as a result of the combination of aerodynamic characteristics, variations with vehicle speed (Mach number), etc.; 2) the atmospheric density, as a result of a combination of the simple model, the parameters in this model, the vehicle's altitude, etc.; and 3) the vehicle's position and velocity.

The equations just described are considered as the nominal or reference model for the evolution of the vehicle's position. The objective of reconstruction is

to estimate errors or deviations from this model as required to agree with independent observations of this motion. In this case, the independent observations are obtained from ground-based radar tracking the vehicle and providing azimuth, elevation, and range relative to its position.

The true position and velocity are represented as the sum of reference values and a perturbation variable:

$$\mathbf{r}^i = \bar{\mathbf{r}}^i + \delta\mathbf{r}^i \quad (10.14)$$

$$\mathbf{v}^i = \bar{\mathbf{v}}^i + \delta\mathbf{v}^i \quad (10.15)$$

It is assumed that equations of motion hold for both the true and reference values. Substituting these equations into Eq. (10.9) and solving for first-order terms in perturbation variables, linear error equations are obtained. Perturbation variable equations are summarized in general state-space form as

$$\frac{d}{dt} \begin{bmatrix} \delta\mathbf{r}^i \\ \delta\mathbf{v}^i \end{bmatrix} = \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} \\ \frac{\partial \mathbf{a}_D^i}{\partial \mathbf{r}^i} + \frac{\partial \mathbf{G}^i}{\partial \mathbf{r}^i} & \frac{\partial \mathbf{a}_D^i}{\partial \mathbf{v}^i} \end{bmatrix} \begin{bmatrix} \delta\mathbf{r}^i \\ \delta\mathbf{v}^i \end{bmatrix} + \begin{bmatrix} 0 \\ \delta\mathbf{a}_D^i + \delta\mathbf{G}^i \end{bmatrix} \quad (10.16)$$

Perturbation variables in atmospheric drag  $\delta\mathbf{a}_D^i$  and gravitation  $\delta\mathbf{G}^i$  can be included as additional state vector elements if they are to be estimated by the Kalman filter. Corresponding estimates represent a departure from the nominal or reference model. The evaluation of each of the partial derivatives follows.

*Higher-order gravitation variation with inertial position.* The indicated partial derivative of gravitation with respect to inertial position should include the  $J_2$  term if a more accurate form of the position error growth associated with the Earth oblate mass is desired. The partial derivative elements of gravitation with respect to position are defined as

$$\frac{\partial \mathbf{G}^i}{\partial \mathbf{r}^i} \equiv \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} \quad (10.17)$$

where

$$A_{11} = -\frac{G_x}{R} + 3 \frac{r_x^2}{R^2} \left\{ \frac{G_x}{R} + \frac{\mu}{R^3} J_2 \frac{r_e^2}{R^2} \left[ 2 \left( 1 - 5 \frac{r_z^2}{R^2} \right) - 1 \right] \right\} \quad (10.18)$$

$$A_{12} = 3 \frac{r_x r_y}{R^2} \left\{ \frac{G_x}{R} + \frac{\mu}{R^3} J_2 \frac{r_e^2}{R^2} \left[ 2 \left( 1 - 5 \frac{r_z^2}{R^2} \right) - 1 \right] \right\} \quad (10.19)$$

$$A_{13} = 3 \frac{r_x r_z}{R^2} \left\{ \frac{G_x}{R} + \frac{\mu}{R^3} J_2 \frac{r_e^2}{R^2} \left[ 2 \left( 1 - 5 \frac{r_z^2}{R^2} \right) + 4 \right] \right\} \quad (10.20)$$

$$A_{21} = A_{12} \quad (10.21)$$

$$A_{22} = -\frac{G_y}{R} + 3\frac{r_y^2}{R^2} \left\{ \frac{G_y}{R} + \frac{\mu}{R^3} J_2 \frac{r_e^2}{R^2} \left[ 2 \left( 1 - 5 \frac{r_z^2}{R^2} \right) - 1 \right] \right\} \quad (10.22)$$

$$A_{23} = 3\frac{r_y r_z}{R^2} \left\{ \frac{G_y}{R} + \frac{\mu}{R^3} J_2 \frac{r_e^2}{R^2} \left[ 2 \left( 1 - 5 \frac{r_z^2}{R^2} \right) + 4 \right] \right\} \quad (10.23)$$

$$A_{31} = A_{13} \quad (10.24)$$

$$A_{32} = A_{23} \quad (10.25)$$

$$A_{33} = -A_{11} - A_{22} \quad (10.26)$$

and  $r_x$ ,  $r_y$ , and  $r_z$  are components of the inertial position vector  $\mathbf{r}^i$ .

**Aerodynamic drag variation with inertial position.** The variation of aerodynamic drag with position includes the combination of density and relative velocity. This is illustrated with the application of the chain differentiation to Eq. (10.10) as

$$\frac{\partial \mathbf{a}_D^i}{\partial \mathbf{r}^i} = -\frac{1}{2} K_D \left[ \frac{\partial \rho}{\partial \mathbf{r}^i} \|\mathbf{v}_{\text{rel}}^i\| \mathbf{v}_{\text{rel}}^i + \rho \frac{\partial \|\mathbf{v}_{\text{rel}}^i\|}{\partial \mathbf{r}^i} \mathbf{v}_{\text{rel}}^i + \rho \|\mathbf{v}_{\text{rel}}^i\| \frac{\partial \mathbf{v}_{\text{rel}}^i}{\partial \mathbf{r}^i} \right] \quad (10.27)$$

where

$$\frac{\partial \rho}{\partial \mathbf{r}^i} \approx -\frac{\rho}{H_s} \begin{bmatrix} r_x \\ R \\ r_y \\ R \\ r_z \\ R \end{bmatrix} \quad (10.28)$$

$$\frac{\partial \|\mathbf{v}_{\text{rel}}^i\|}{\partial \mathbf{r}^i} = \frac{1}{\|\mathbf{v}_{\text{rel}}^i\|} (\boldsymbol{\omega}_{i/e}^i \times) (\boldsymbol{\omega}_{i/e}^i \times) \mathbf{r}^i \quad (10.29)$$

$$\frac{\partial \mathbf{v}_{\text{rel}}^i}{\partial \mathbf{r}^i} = -(\boldsymbol{\omega}_{i/e}^i \times) \quad (10.30)$$

**Aerodynamic drag variation with inertial velocity.** Because density is not a function of velocity, the partial derivative of aerodynamic drag becomes

$$\frac{\partial \mathbf{a}_D^i}{\partial \dot{\mathbf{r}}^i} = -\frac{1}{2} K_D \rho \left[ \frac{\partial \|\mathbf{v}_{\text{rel}}^i\|}{\partial \dot{\mathbf{r}}^i} \mathbf{v}_{\text{rel}}^i + \|\mathbf{v}_{\text{rel}}^i\| \frac{\partial \mathbf{v}_{\text{rel}}^i}{\partial \dot{\mathbf{r}}^i} \right] \quad (10.31)$$

where

$$\frac{\partial \|\mathbf{v}_{\text{rel}}^i\|}{\partial \dot{\mathbf{r}}^i} = \frac{1}{\|\mathbf{v}_{\text{rel}}^i\|} \dot{\mathbf{r}}^T \quad (10.32)$$

$$\frac{\partial \mathbf{v}_{\text{rel}}^i}{\partial \dot{\mathbf{r}}^i} = I_{3 \times 3} \quad (10.33)$$

### 10.3.3 Radar Position Observations

The vehicle's position is monitored with ground-based tracking radar. Because the radar is fixed to the Earth's surface, its outputs, based on a local-level coordinate system, are in the form of a relative position as azimuth, elevation, and range. The radar's outputs must be related to the Earth-centered inertial coordinate system used to express the vehicle's position/velocity evolution.

Consider the geometry shown in Fig. 10.14. The relative position between the radar site and the vehicle is expressed as

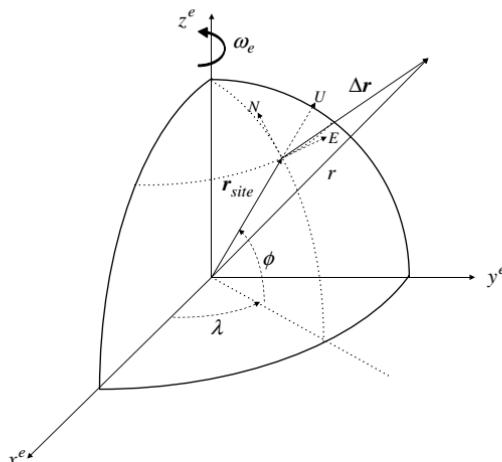
$$\Delta \mathbf{r}^e = C_i^e \mathbf{r}^i - \mathbf{r}_{\text{site}}^e \quad (10.34)$$

Or, in terms of the local geographic frame,

$$\Delta \mathbf{r}^g = C_e^g (C_i^e \mathbf{r}^i - \mathbf{r}_{\text{site}}^e) \quad (10.35)$$

Following the sequence of rotations of longitude  $\lambda$  and latitude  $\phi$  shown in Fig. 10.14 and a final transformation to align the  $x$ - $y$ - $z$  axes with east–north–up, respectively, yields the following form for the  $C_e^g$  transformation matrix:

$$C_e^g = \begin{bmatrix} -s\lambda & c\lambda & 0 \\ -s\phi c\lambda & -s\phi c\lambda & c\phi \\ c\phi c\lambda & c\phi s\lambda & s\phi \end{bmatrix} \quad (10.36)$$



**Fig. 10.14** Relative position with respect to ground radar site.

The  $C_i^e$  transformation matrix is expressed as

$$C_i^e = \begin{bmatrix} c\omega_{i/e}t & s\omega_{i/e}t & 0 \\ -s\omega_{i/e}t & c\omega_{i/e}t & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10.37)$$

where  $t$  = time relative to the beginning of motion/tracking.

Components of the relative position vector in Eq. (10.35) are ordered as east–north–up. Defining the radar azimuth as positive clockwise relative to north, the following equations express the radar's azimuth  $Az$ , elevation  $El$ , and range  $Rng$  outputs:

$$Az = \tan^{-1}\left(\frac{\Delta r_x}{\Delta r_y}\right) \quad (10.38)$$

$$El = \tan^{-1}\left(\frac{\Delta r_z}{\sqrt{\Delta r_x^2 + \Delta r_y^2}}\right) \quad (10.39)$$

and

$$Rng = \sqrt{\Delta r_x^2 + \Delta r_y^2 + \Delta r_z^2} \quad (10.40)$$

where  $\Delta r_x$ ,  $\Delta r_y$ , and  $\Delta r_z$  are the components of the relative position vector  $\Delta \mathbf{r}^e$ .

#### 10.3.4 Radar Position Observation Linearizations

Variations of Eqs. (10.38–10.40) with respect to states are developed next.

*Azimuth variation with inertial position.* The partial derivative of azimuth is expressed as

$$\frac{\partial Az}{\partial \mathbf{r}^i} = \frac{\partial Az}{\partial \Delta \mathbf{r}^g} \frac{\partial \Delta \mathbf{r}^g}{\partial \mathbf{r}^i} \quad (10.41)$$

The second of the partial derivatives on the RHS of this equation is used here and later. This partial derivative is obtained from Eq. (10.35) as

$$\frac{\partial \Delta \mathbf{r}^g}{\partial \mathbf{r}^i} = C_e^g C_i^e \quad (10.42)$$

The first partial derivative is obtained from Eq. (10.38) as

$$\frac{\partial Az}{\partial \Delta \mathbf{r}^g} = \left[ \frac{\Delta r_y}{(\Delta r_x^2 + \Delta r_y^2)} \quad \frac{-\Delta r_x}{(\Delta r_x^2 + \Delta r_y^2)} \quad 0 \right] \quad (10.43)$$

*Elevation variation with inertial position.* The partial derivative of elevation is

$$\frac{\partial El}{\partial \mathbf{r}^i} = \frac{\partial El}{\partial \Delta \mathbf{r}^g} \frac{\partial \Delta \mathbf{r}^g}{\partial \mathbf{r}^i} \quad (10.44)$$

The first partial derivative is obtained from Eq. (10.39) as

$$\frac{\partial El}{\partial \Delta \mathbf{r}^g} = \begin{bmatrix} -\Delta r_x \Delta r_z \\ R^2 \sqrt{\Delta r_x^2 + \Delta r_y^2} & -\Delta r_y \Delta r_z \\ R^2 \sqrt{\Delta r_x^2 + \Delta r_y^2} & \frac{\sqrt{\Delta r_x^2 + \Delta r_y^2}}{R^2} \end{bmatrix} \quad (10.45)$$

The second derivative was obtained earlier as Eq. (10.42).

*Range variation with inertial position.* The partial derivative of range is

$$\frac{\partial Rng}{\partial \mathbf{r}^i} = \frac{\partial Rng}{\partial \Delta \mathbf{r}^g} \frac{\partial \Delta \mathbf{r}^g}{\partial \mathbf{r}^i} \quad (10.46)$$

The first partial derivative is obtained from Eq. (10.40) as

$$\frac{\partial Rng}{\partial \Delta \mathbf{r}^g} = \begin{bmatrix} \Delta r_x \\ R & \Delta r_y \\ R & \Delta r_z \\ R \end{bmatrix} \quad (10.47)$$

### 10.3.5 Measurements

Measurement equations can be summarized by the following:

$$\begin{aligned} \mathbf{z} &\equiv \begin{bmatrix} Az_m - \hat{Az} \\ El_m - \hat{El} \\ Rng_m - \hat{Rng} \end{bmatrix} \\ &= H\mathbf{x} + \mathbf{v} \\ &= \begin{bmatrix} \frac{\partial Az}{\partial \mathbf{r}^i} & 0^T \\ \frac{\partial El}{\partial \mathbf{r}^i} & 0^T \\ \frac{\partial Rng}{\partial \mathbf{r}^i} & 0^T \end{bmatrix} \begin{bmatrix} \delta \mathbf{r}^i \\ \delta \mathbf{v}^i \end{bmatrix} + \mathbf{v} \end{aligned} \quad (10.48)$$

where  $Az_m$ ,  $El_m$ , and  $Rng_m$  are measured values and  $\hat{Az}$ ,  $\hat{El}$ , and  $\hat{Rng}$  are estimated values, and their differences are related by the matrix/state vector product indicated.

### 10.3.6 Application to Sounding Rocket Trajectory Reconstruction

The preceding equations are applied to the reconstruction of a sounding rocket's trajectory based on the observed motion via a ground-tracking radar. Shown in Figs. 10.15 and 10.16 are a typical meteorological sounding rocket and its launcher. In this application, actual data used are for a rocket similar to that shown in Fig. 10.15, a PWN-11D. This sounding rocket consists of a second-stage unpowered dart, within which is the payload. The payload is a ram-air inflated ballute that is released at approximately 120 s into its flight. The dart is lofted by the vehicle's first-stage rocket motor. For the configuration shown in Fig. 10.15, the ballistic coefficient  $K_D$  in Eq. (10.10) varies from less than one to several thousand for its payload—the ram-air inflated ballute.

Ground radar tracking data are intermittent for flight-test data used. Shown in Figs. 10.17 and 10.18 are elevation and range data measured during the first 500 s of flight. During the ascent flight phase, dropouts dominate the test data. This also can occur in descent phase. The linearized Kalman filter implemented for this illustration would have difficulty providing an accurate trajectory reconstruction if it were not for the reference profile generated by numerically integrating Eq. (10.9). Relative to this reference trajectory, filter algorithm estimates' corrections are then applied external to the estimation process to provide whole value estimates.

The primary objective is to estimate local winds while the ballute is descending to the lower atmosphere. Consistent with this objective, the aerodynamic drag equation is reasonable for this phase; however, it is not representative for

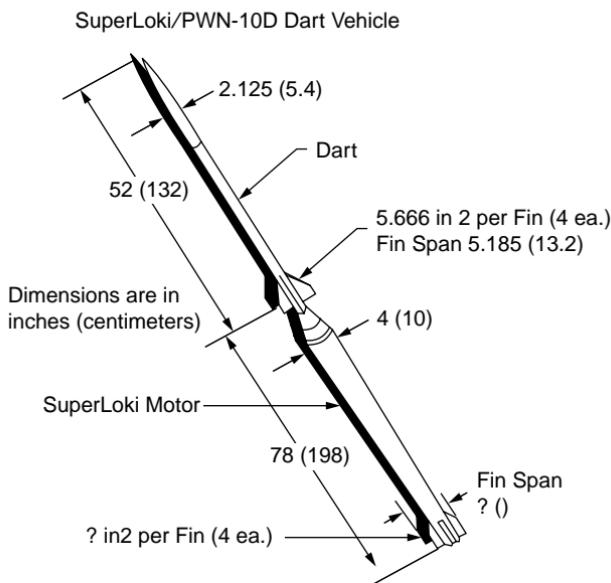
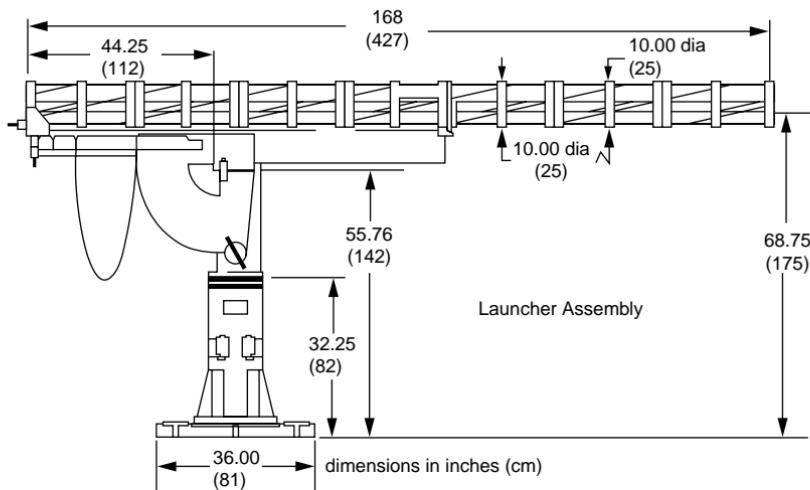


Fig. 10.15 Typical meteorological sounding rocket.

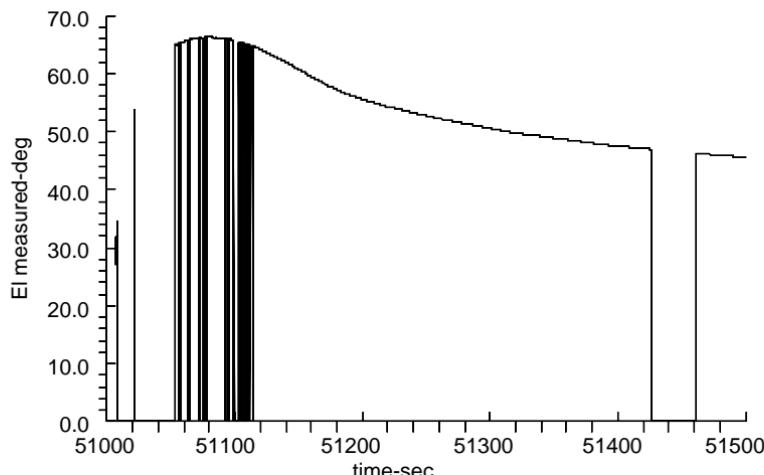
## Launcher



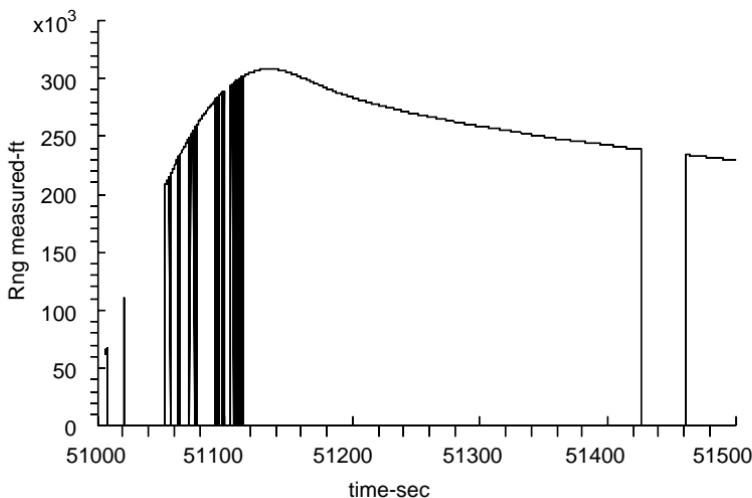
**Fig. 10.16** Launcher for sounding rocket in Fig. 10.15.

powered and dart-coasting phases of the flight. To address this limitation, the aerodynamic drag perturbation variable  $\delta a_D^i$  in Eq. (10.16) is modeled as a time-correlated random process to allow for a time-varying characteristic to this variable to account for the differences in the vehicle's aerodynamic characteristics modeled by Eq. (10.10) for ascent and descent phases.

Reconstructed trajectory profiles are shown in Figs. 10.19 and 10.20 for altitude and horizontal positions, respectively. These figures show that the trajectory is nearly ballistic until the ballute descends to approximately 200,000 ft. At



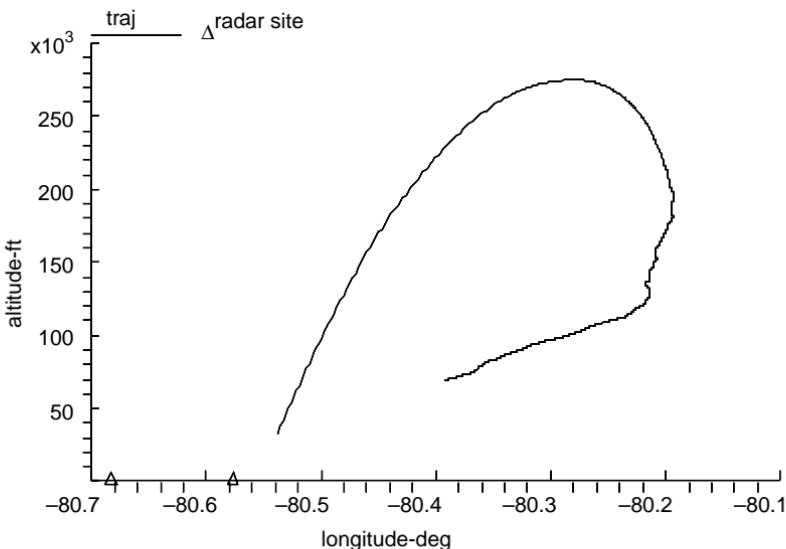
**Fig. 10.17** Ground radar measured elevation.



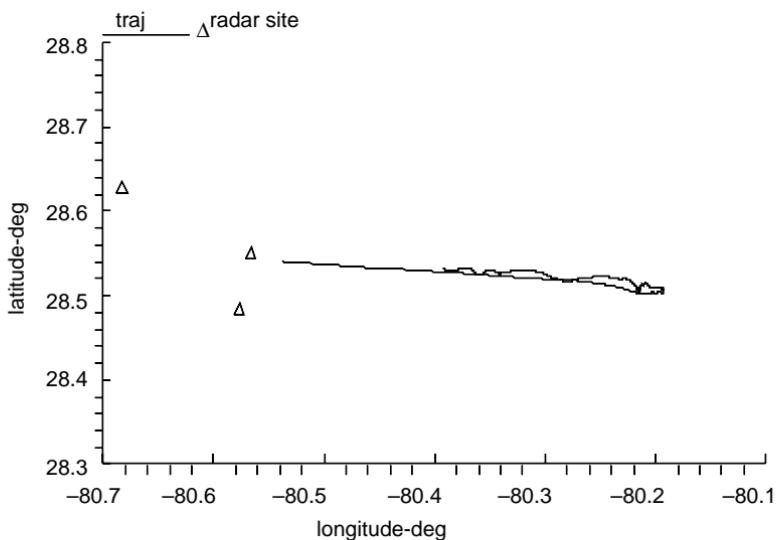
**Fig. 10.18** Ground radar measured range.

approximately 120,000 ft and below, the ballute can be seen to be affected by the eastward wind. From its launch point, the sounding rocket was fired almost directly into the wind. Also shown in these figures are ground radar locations. The northwest radar's azimuth, elevation, and range data were used to generate these reconstructions.

The estimated total relative speed for the entire trajectory is shown in Fig. 10.21. At the 120-s point, a perturbation in the estimated speed is seen.



**Fig. 10.19** Reconstructed altitude profile.

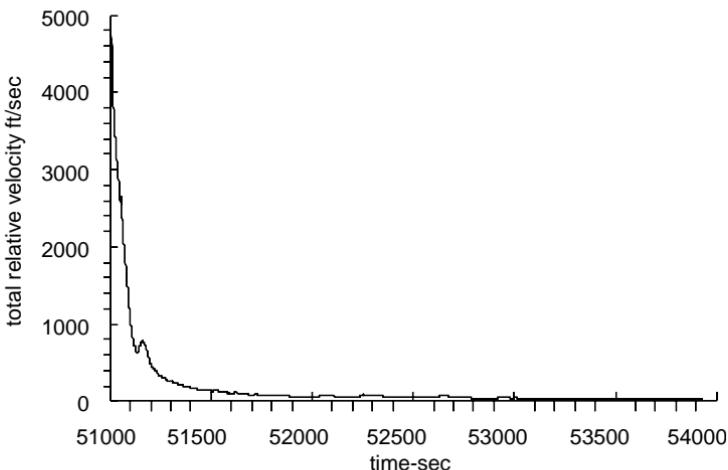


**Fig. 10.20** Reconstructed horizontal plane position.

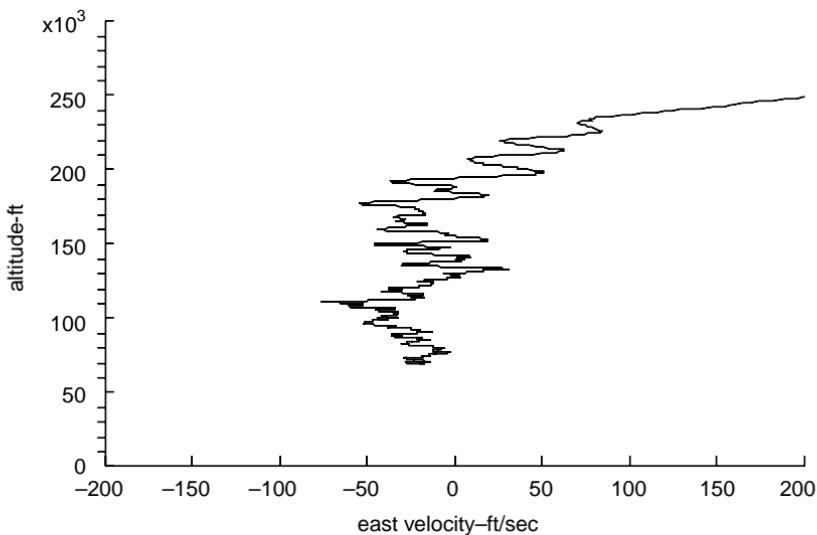
This is likely the result of a combination of factors: 1) the change in aerodynamic configuration from dart to ballute and the inadequacy of the aerodynamic drag model to represent this change instantaneously and/or 2) ground-based radar's ability to track the dart during the staging process.

The estimated local east velocity is shown in Fig. 10.22. From Fig. 10.19, velocities below 120,000 ft are representative of wind conditions. Oscillations in the local velocity are likely caused by rotations of the ballute during descent.

A quality measure of the preceding estimates is the measurement residuals. Shown in Figs. 10.23 and 10.24 are radar elevation and range measurement

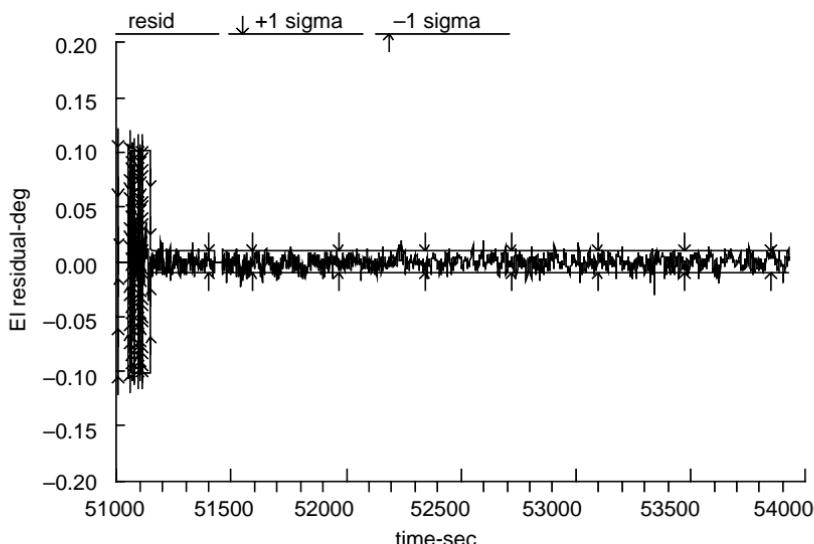


**Fig. 10.21** Reconstructed total relative velocity.

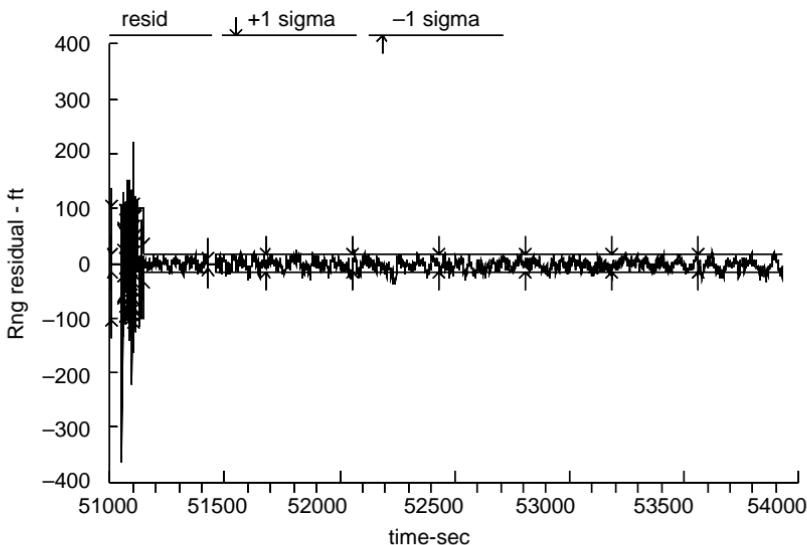


**Fig. 10.22** Reconstructed east velocity profile.

residuals, respectively. At the 120-s point, radar measurement variances were changed to correspond to an apparent change in the radar's operating mode. At about the same time, the vehicle's aerodynamic configuration changed from a dart to a ballute. These residuals appear reasonable, with little bias seen and most of the residuals lying within the  $1-\sigma$  upper and lower bounds. The variance



**Fig. 10.23** Ground radar elevation measurement residual.



**Fig. 10.24** Ground radar range measurement residual.

assumed for radar tracking errors, that is, azimuth, elevation, and range, is a tradeoff between a desire for smoothness of estimated velocity data shown in Fig. 10.22 and recognition that measurements are nonlinearly related to the filter's states. For example, if smaller residuals were used, the velocity profile would appear more erratic than that shown. During the periods that radar data were not available, measurements were excluded using a function similar to the residual-editing features described in Chapter 8. In this case, when measurement data were exactly zero, measurements were excluded. This is reasonable because the likelihood that actual measurements are exactly zero is very small. The facility that provided the recorded data listed the data record as zero when radar data were not available.

## 10.4 Chapter Summary

Three techniques for trajectory reconstruction for flight-test evaluations were presented. The first used optical tracking data to reconstruct an aircraft's position time history. This technique computes a least-squares estimate of the change in position based on two line-of-sight observations from ground-based tracking cameras. There are no dynamics associated with this technique, and only the trajectory is reconstructed (estimated).

The second technique, based on TACAN ranging integrated with an INU, was used to reconstruct navigation states. This technique combines the inertial-navigation system under test with TACAN ranging in a Kalman filter to obtain corrected position, velocity, and attitude navigation error states.

Finally, a reconstruction based on a vehicle's translational dynamics was presented. In this application, reference models are equations of motion for a

sounding rocket, which includes applied aerodynamic and gravitational forces. Ground-based radar observations are used in a Kalman filter algorithm to reconstruct the rocket's position and velocity.

In the latter two examples, the reference dynamical process, INU data for the second, and vehicle translational dynamics in the third, provides a smooth transition between observation times. This smooth transition permits the use of intermittent observations within the Kalman filter algorithm.

## Inertial Navigation System Ground Alignment

In this chapter, a third application of the elements from Part 1, inertial navigation system ground alignment, is presented. The principal objective is to establish the navigation system's initial attitudes (roll, pitch, and heading). The more difficult objective to accomplish is establishing the system's initial azimuth/heading. This illustration includes the following: initial coarse alignment and resulting errors, fine-alignment Kalman filter, and simulated ground fine-alignment.

Ground alignment is accomplished in stages, with increasing levels of accuracy. The initial phase, coarse alignment or leveling, provides an estimate of initial attitudes. In the final phase, fine-alignment, a Kalman filter is used to refine the alignment and estimate inertial sensor errors prior to flight.

Coarse alignment takes advantage of a known position. Using this assumed known position, the Earth's rotation and gravity are computed. Comparing computed rotation rates and gravity to sensed rates and acceleration, estimates of initial attitudes are computed. This comparison requires that the inertial navigation unit's sensors be of sufficient quality to yield a reasonable estimate of initial attitude. The effect of sensor quality on initial alignment accuracy is presented.

In addition to the gyro-compassing, other alignment modes include stored heading and best-available true heading. Stored heading alignment uses the heading stored from the navigation system's previous operation, assuming that the navigation unit has not been moved. The magnetic heading, compensated for magnetic variation, is a candidate source for the best-available true heading. The heading established by these sources is used to initialize the  $C_b^n$  matrix. Some knowledge of heading, although imprecise, permits the use of small attitude error models in a fine-alignment Kalman filter implementation (see Chapter 13).

The final fine-alignment phase is accomplished using a Kalman filter algorithm. Initial attitudes and sensor error estimates are refined during this final alignment stage. Assuming a stationary vehicle, the actual velocity components in the navigation frame are zero. The east component of rate output for a geographic-frame-referenced navigation frame should also be zero. Velocity and rate components are used as measurements in the fine-alignment filter. This fine-alignment Kalman filter processing will be illustrated using simulated inertial navigation unit data.

## 11.1 Initial Coarse Alignment and Resulting Errors

This technique, from [1] uses inherent properties of the Earth's gravity and rotation to determine the inertial navigation system's initial attitude.

### 11.1.1 Stationary Alignment

Assume that the following are available as outputs from a strap-down inertial navigation system:

$$\mathbf{g}^b = C_n^b \mathbf{g}^n \quad (11.1)$$

$$\boldsymbol{\omega}_{i/e}^b = C_n^b \boldsymbol{\omega}_{i/e}^n \quad (11.2)$$

Then, the following matrix product is formed, using these vectors as the matrix's columns:

$$[\mathbf{g}^b \quad \boldsymbol{\omega}_{i/e}^b \quad \mathbf{g}^b \times \boldsymbol{\omega}_{i/e}^b] = C_n^b [\mathbf{g}^n \quad \boldsymbol{\omega}_{i/e}^n \quad \mathbf{g}^n \times \boldsymbol{\omega}_{i/e}^n] \quad (11.3)$$

Or, transposing yields

$$[\mathbf{g}^b \quad \boldsymbol{\omega}_{i/e}^b \quad \mathbf{g}^b \times \boldsymbol{\omega}_{i/e}^b]^T = [\mathbf{g}^n \quad \boldsymbol{\omega}_{i/e}^n \quad \mathbf{g}^n \times \boldsymbol{\omega}_{i/e}^n]^T C_b^n \quad (11.4)$$

Inverting the  $3 \times 3$   $[ ]^T$  matrix on the right-hand side and premultiplying both sides with the results, the following is obtained:

$$C_b^n = [\mathbf{g}^n \quad \boldsymbol{\omega}_{i/e}^n \quad \mathbf{g}^n \times \boldsymbol{\omega}_{i/e}^n]^{-T} [\mathbf{g}^b \quad \boldsymbol{\omega}_{i/e}^b \quad \mathbf{g}^b \times \boldsymbol{\omega}_{i/e}^b]^T \quad (11.5)$$

This matrix can be solved to produce an initial determination of the  $C_b^n$  direction cosine matrix.

To illustrate the application of this technique, assume the preceding navigation frame is a local-level geographic *N-E-D* frame. The gravity and Earth rotation vectors become

$$\mathbf{g}^n = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (11.6)$$

and

$$\boldsymbol{\omega}_{i/e}^n = \begin{bmatrix} \omega_{i/e} \cos \phi \\ 0 \\ -\omega_{i/e} \sin \phi \end{bmatrix} \quad (11.7)$$

The cross product of these two vectors is given by

$$\mathbf{g}^n \times \boldsymbol{\omega}_{i/e}^n = \begin{bmatrix} 0 \\ g\omega_{i/e} \cos \phi \\ 0 \end{bmatrix} \quad (11.8)$$

The matrix inverse/transpose in Eq. (11.5) becomes

$$[ ]^{-T} = \begin{bmatrix} \frac{\tan \phi}{g} & \frac{1}{\omega_{i/e} \cos \phi} & 0 \\ 0 & 0 & \frac{1}{g\omega_{i/e} \cos \phi} \\ \frac{1}{g} & 0 & 0 \end{bmatrix} \quad (11.9)$$

If the alignment latitude and local gravity are known, and if the system's sensor outputs were without error, this equation is used in Eq. (11.5) to compute the initial  $C_b^n$  direction cosine matrix. The effect of sensor errors on the alignment accuracy is examined next.

### 11.1.2 Stationary Alignment Errors

A stationary strap-down inertial navigation unit provides sensed gravity and Earth rotations by way of its accelerometers and gyro. These sensors' outputs contain error; therefore,  $C_b^n$  computed as just outlined contains error, and this computed matrix, with error, is represented with an overbar and expressed as

$$\bar{C}_b^n = [\mathbf{g}^n \quad \boldsymbol{\omega}_{i/e}^n \quad \mathbf{g}^n \times \boldsymbol{\omega}_{i/e}^n]^{-T} [\bar{\mathbf{f}}^b \quad \bar{\boldsymbol{\omega}}_{i/b}^b \quad \bar{\mathbf{f}}^b \times \bar{\boldsymbol{\omega}}_{i/e}^b]^T \quad (11.10)$$

Sensed specific force and gyro rate errors are represented as the perturbed variables used earlier:

$$\bar{\mathbf{f}}^b = \mathbf{g}^b + \delta \mathbf{f}^b \quad (11.11)$$

$$\bar{\boldsymbol{\omega}}_{i/b}^b = \boldsymbol{\omega}_{i/e}^b + \delta \boldsymbol{\omega}^b \quad (11.12)$$

Assuming small attitude errors, the computed  $C_b^n$  matrix is represented as before:

$$\bar{C}_b^n = [\mathbf{I} - (\boldsymbol{\phi} \times)] C_b^n \quad (11.13)$$

Equating this expression with Eq. (11.10) yields

$$[\mathbf{I} - (\boldsymbol{\phi} \times)] C_b^n = [ ]^{-T} \left\{ [\mathbf{g}^b \quad \boldsymbol{\omega}_{i/e}^b \quad \mathbf{g}^b \times \boldsymbol{\omega}_{i/e}^b]^T + [\delta \mathbf{f}^b \quad \delta \boldsymbol{\omega}^b \quad \delta(\mathbf{f}^b \times \boldsymbol{\omega}^b)]^T \right\} \quad (11.14)$$

Subtracting Eq. (11.5) from this equation yields

$$-(\boldsymbol{\phi} \times) C_b^n = [ ]^{-T} [\delta f^b \quad \delta \omega^b \quad \delta(f^b \times \omega^b)]^T \quad (11.15)$$

This equation can be used to relate navigation frame tilt errors to body-referenced sensor errors for any navigation-frame mechanization.

Again, to illustrate this relationship, the navigation frame is assumed to be a geographic frame. Frame misalignment is described by the following cross product skew-symmetric matrix equivalent:

$$-(\boldsymbol{\phi} \times) = \begin{bmatrix} 0 & \phi_d & -\phi_e \\ -\phi_d & 0 & \phi_n \\ \phi_e & -\phi_n & 0 \end{bmatrix} \quad (11.16)$$

From Eq. (11.7), the east component of rate is zero if the inertial navigation unit is perfectly aligned with the geographic frame. However, as a result of a misalignment, there is error in sensed rate. This error is represented as

$$\bar{\omega}_{i/e}^b = \bar{C}_n^b \omega_{i/e}^n$$

or

$$\omega_{i/e}^b + \delta \omega^b = C_n^b [I + (\boldsymbol{\phi} \times)] \omega_{i/e}^n \quad (11.17)$$

The error rate becomes

$$\begin{aligned} \delta \omega^b &= C_n^b (\boldsymbol{\phi} \times) \omega_{i/e}^n \\ &= -C_n^b \begin{bmatrix} 0 & \phi_d & -\phi_e \\ -\phi_d & 0 & \phi_n \\ \phi_e & -\phi_n & 0 \end{bmatrix} \begin{bmatrix} \omega_{i/e} \cos \phi \\ 0 \\ -\omega_{i/e} \sin \phi \end{bmatrix} \\ &= C_n^b \begin{bmatrix} -\phi_e \omega_{i/e} \sin \phi \\ \phi_d \omega_{i/e} \cos \phi + \phi_n \omega_{i/e} \sin \phi \\ -\phi_e \omega_{i/e} \cos \phi \end{bmatrix} \quad (11.18) \end{aligned}$$

The east rate error component is

$$\delta \omega_y^n = \phi_d \omega_{i/e} \cos \phi + \phi_n \omega_{i/e} \sin \phi \quad (11.19)$$

Adjustments (gyro-compassing) in the azimuth error  $\phi_d$  reduce the  $\delta\omega_y^n$  rate error in this equation. This rate error is used later as a measurement in a Kalman filter during fine-alignment.

### Example 11.1 Ground Alignment Accuracy Dependency on Sensors' Errors

To simplify the problem for the purposes of this example, assume that the navigation system is nominally aligned with the geographic frame such that

$$C_b^n \equiv I$$

Substituting Eqs. (11.6–11.9) into Eq. (11.15) and equating the corresponding components on the left and right sides yield the following for the navigation frame misalignments:

$$\begin{aligned}\phi_n &= -\frac{\delta f_y}{g} \\ \phi_e &= \frac{\delta f_x}{g} \\ \phi_d &= \frac{\delta f_y}{g} \tan \phi + \frac{\delta \omega_y}{\omega_{i/e} \cos \phi}\end{aligned}$$

This result demonstrates that misalignment errors are introduced as a result of accelerometer and gyro errors when using this initial alignment technique and that the magnitude of azimuth alignment error is dependent on the alignment latitude.

## 11.2 Fine-Alignment Kalman Filter

The fine-alignment phase(s) begins assuming a direction cosine matrix determined by either the initial coarse alignment technique already discussed or by using as initialization data a stored heading or best-available true heading. The initial estimate for the  $C_b^n$  matrix is then refined by using a Kalman filter. Measurements processed in this filter can include velocity, Earth rate, etc.

To illustrate fine alignment, the geographic frame is again used as the navigation frame. The general form for linearized error equations is presented in Chapter 5. Inertial sensor errors to be included are accelerometer and gyro errors. With the assumption of a stationary inertial navigation unit, the reduced state error dynamics model used in the fine-alignment Kalman filter is summarized in Fig. 11.1. Of the possible inertial sensor errors (see Sec. 6.2), accelerometer and gyro time-correlated errors are included as additional states in the alignment Kalman filter's state vector. The process noise, indicated as the product  $Gw$ , contains the random errors in velocity and attitude error states for unmodeled accelerometer and gyro errors, respectively, and is used for tuning the alignment filter. Also included in this process-noise vector are the terms

$$\begin{bmatrix}
 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & -2\Omega_d & 0 & -g & 0 & g & 0 & C_b^g_{11} & C_b^g_{12} & C_b^g_{13} & 0 & 0 & 0 \\
 \delta r_n & \delta r_e & \delta v_n & \delta v_e & 2\Omega_d & 0 & 0 & 0 & C_b^g_{21} & C_b^g_{22} & C_b^g_{23} & 0 & 0 & 0 & 0 \\
 \delta \dot{r}_n & \delta \dot{r}_e & \delta \dot{v}_n & \delta \dot{v}_e & -2\Omega_d & 0 & -g & 0 & 0 & 0 & 0 & 0 & C_b^g_{11} & C_b^g_{12} & C_b^g_{13} \\
 \phi_n & \phi_e & \phi_d & \phi_u & 1/R & 0 & \Omega_d & 0 & 0 & 0 & 0 & 0 & C_b^g_{21} & C_b^g_{22} & C_b^g_{23} \\
 \dot{\phi}_n & \dot{\phi}_e & \dot{\phi}_d & \dot{\phi}_u & -1/R & 0 & -\Omega_d & 0 & 0 & 0 & 0 & 0 & C_b^g_{31} & C_b^g_{32} & C_b^g_{33} \\
 \ddot{\phi}_n & \ddot{\phi}_e & \ddot{\phi}_d & \ddot{\phi}_u & -t\phi/R & 0 & -\Omega_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \frac{d}{dt} & = & ab_x & ab_y & ab_z & 0 & 0 & 0 & 0 & -1/\tau_a & 0 & 0 & 0 & 0 & 0 & 0 \\
 \frac{d}{dt} & ab_x & ab_y & ab_z & gB_x & 0 & 0 & 0 & 0 & 0 & -1/\tau_a & 0 & 0 & 0 & 0 & 0 \\
 \frac{d}{dt} & gB_x & gB_y & gB_z & gB_{-z} & 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_a & 0 & 0 & 0 & 0 \\
 \frac{d}{dt} & gB_y & gB_z & gB_{-x} & gB_{-x} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_g & 0 & 0 & 0 \\
 \frac{d}{dt} & gB_z & gB_{-x} & gB_{-y} & gB_{-y} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/\tau_g & 0
 \end{bmatrix} + Gw$$

Fig. 11.1 Reduced state error dynamics for fine alignment.

$$H = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \Lambda \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \Lambda \\ 0 & 0 & 0 & 0 & \Omega_d & 0 & -\Omega_n & 0 & \Lambda \end{bmatrix}$$

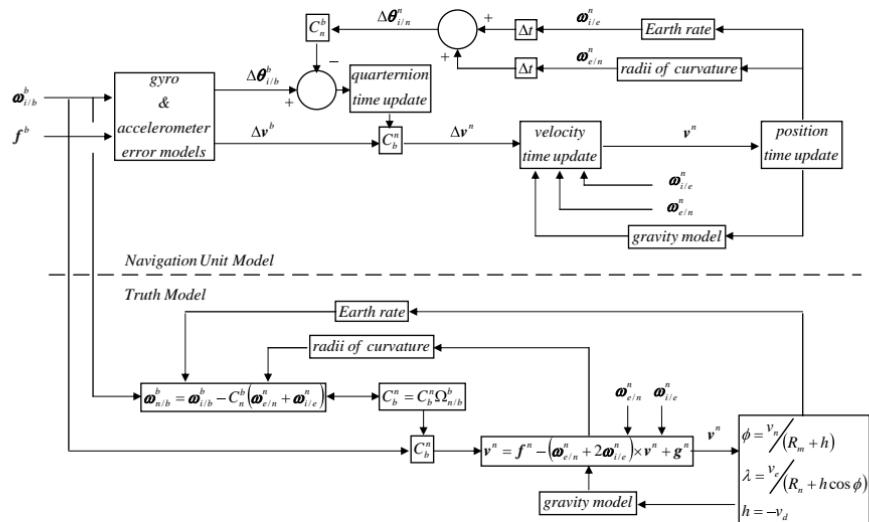
**Fig. 11.2 Measurement matrix for fine alignment.**

for time-correlated accelerometer error states (see Sec. 6.2). The  $H$  measurement matrix shown in Fig. 11.2 includes north and east velocities and geographic frame east rate measurements:

### 11.3 Simulated Ground Fine Alignment

To illustrate the Kalman filter fine alignment, data are synthetically generated from a simulated inertial navigation unit (INU). The simulation incorporates models for the navigation unit's navigation equations, as corrupted by initialization and sensor errors, and navigation equations representing a true navigation solution with no initialization or sensor errors. The numerical integration simulated for navigation's attitude equations for generating the  $C_b^n$  direction cosine matrix is based on using quaternions (see Sec. 2.5). Both sets of equations use true accelerations and rates provided by a driver program, as shown in Fig. 11.3. The driver program generates accelerations and rates representative of a simulated dynamic environment. For this illustration, the simulated environment is a stationary INU.

The INU simulated is an attitude heading reference system (AHRS). An AHRS is a moderate quality strap-down INU. Its sensors are mounted on a turn-table that rotates in azimuth during the alignment phase. Initial INU attitudes, as



**Fig. 11.3 Simulation program flow.**

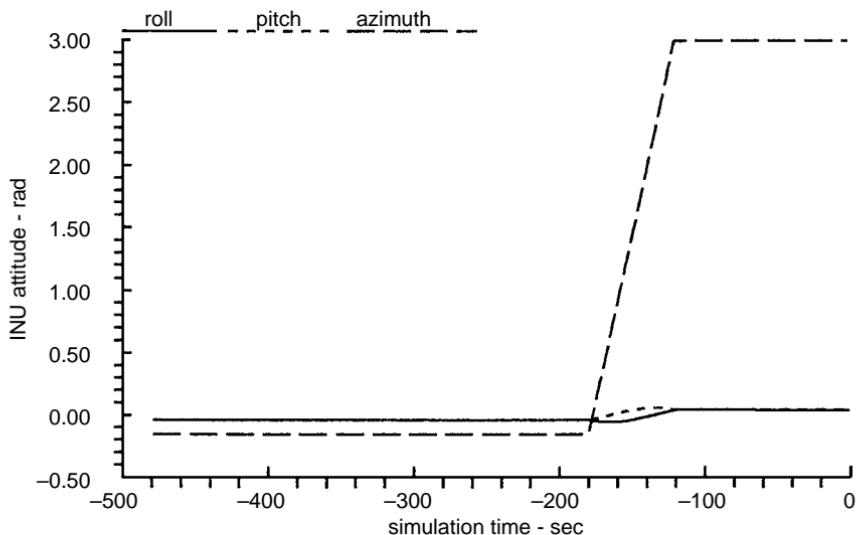


Fig. 11.4 Simulated INU attitudes.

well as INU instrument errors, are selected randomly by the simulation program. Shown in Figs. 11.4 and 11.5 are the INU's attitudes and velocities during the simulated eight-minute alignment period, which continues until  $t = 0$ . If the INU were without error, the velocity components would be zero. Shown in Figs. 11.6 and 11.7 are simulated INU north and east velocities, the alignment filter's estimated velocities with their  $\pm 1\sigma$  uncertainty bounds.

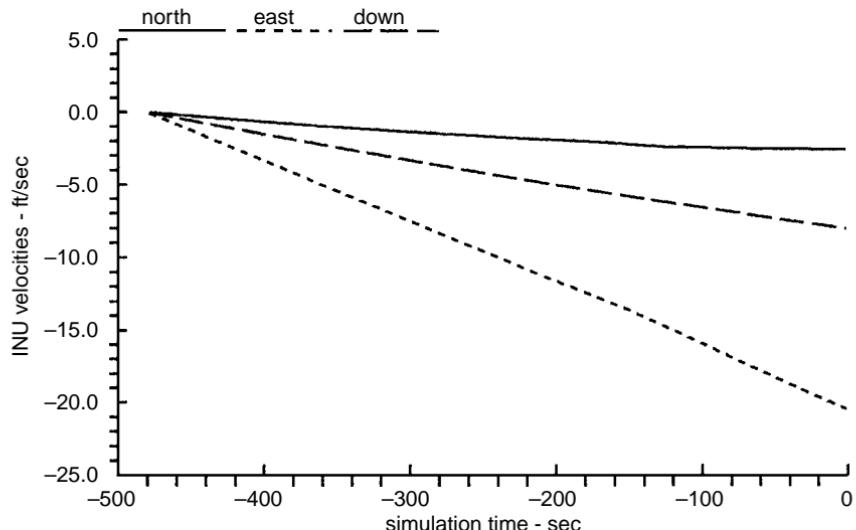


Fig. 11.5 Simulated INU velocities.

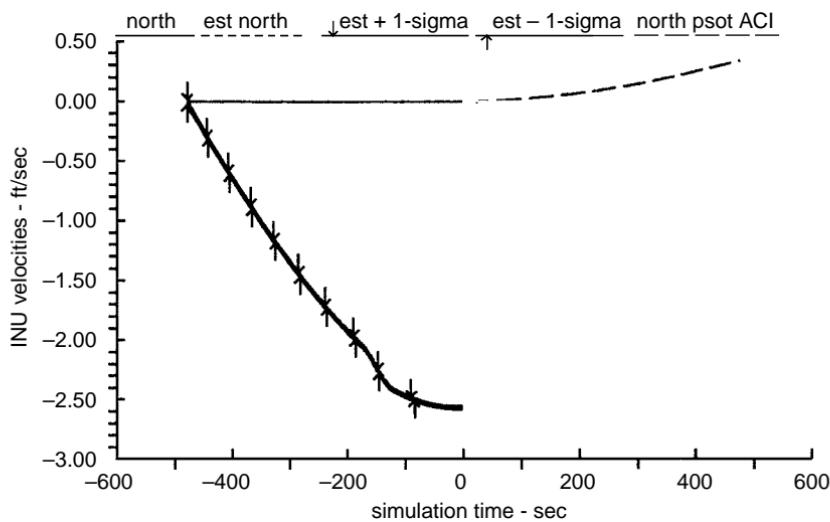


Fig. 11.6 North velocity simulated alignment.

The alignment simulation is allowed to continue beyond  $t = 0$  time, which represents the completion of the alignment, calibration, and initialization (ACI) phase. If the alignment filter improved estimates of initial attitudes and sensor errors, the rate of growth of the velocity errors should be reduced from the initial coarse alignment values. At the designated  $t = 0$ , estimated tilt and instrument errors are used to compensate further outputs from the INU model. The navigation phase, post-ACI, is initialized with the same position and zero

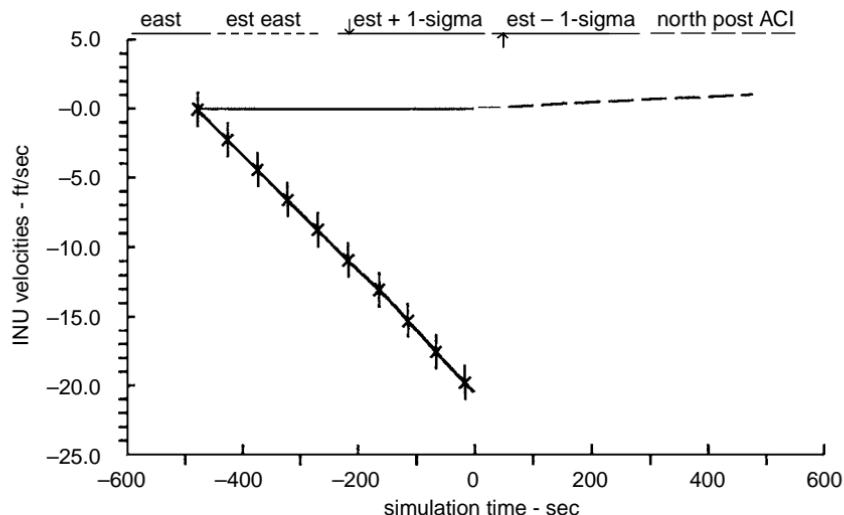
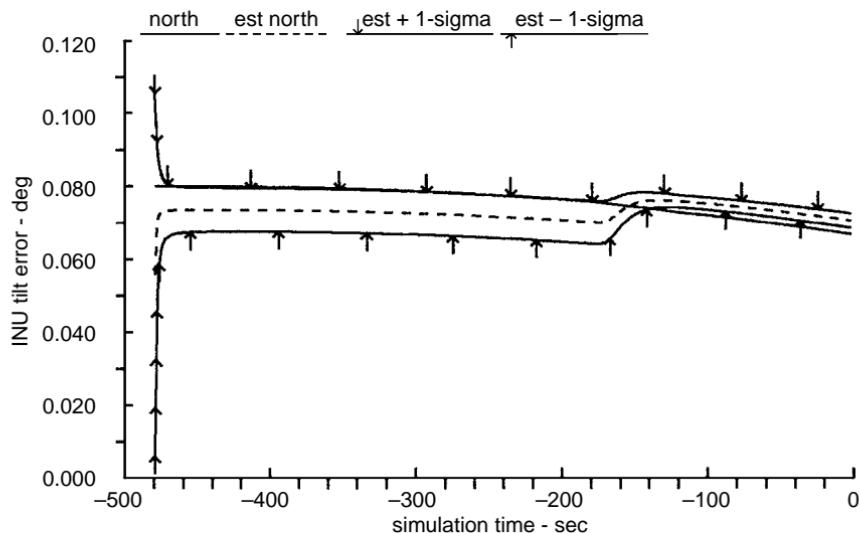


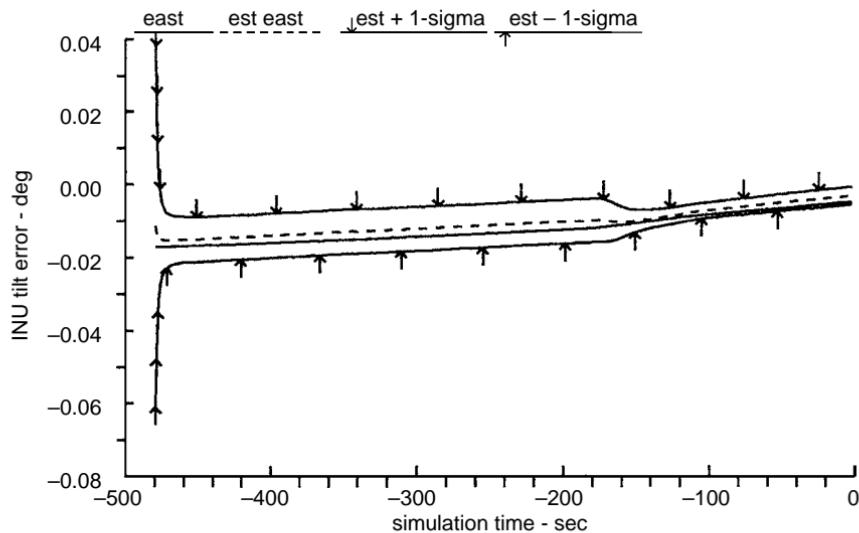
Fig. 11.7 East velocity simulated alignment.



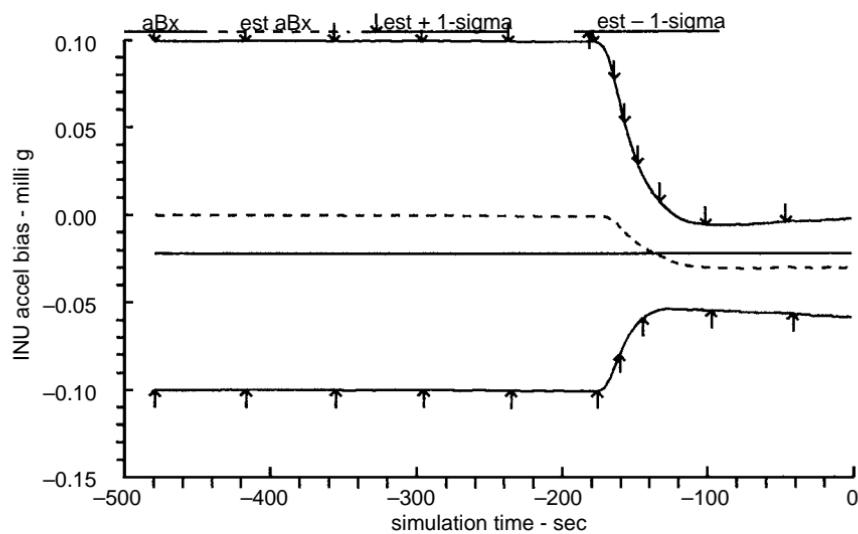
**Fig. 11.8** North tilt simulated alignment.

velocity. Shown in Figs. 11.6 and 11.7 also are the velocity components post-ACI. After  $t = 0$ , the velocity error's growth rates are shown to be reduced. These results indicate that the fine-alignment Kalman filter's estimates improve the INU's attitude for post-ACI operations.

During fine alignment, the Kalman filter provides estimated tilt error, azimuth error, accelerometer, and gyro biases. Shown in Figs. 11.8 and 11.9 are actual

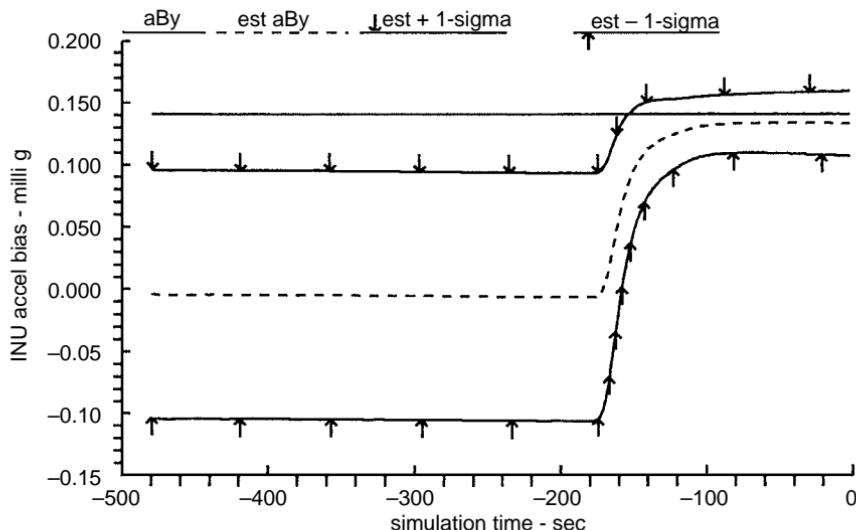


**Fig. 11.9** East tilt simulated alignment.

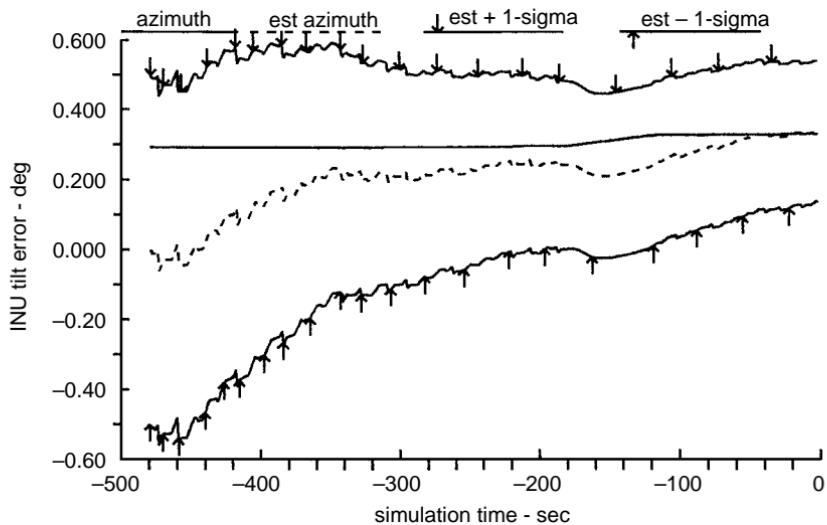


**Fig. 11.10 Alignment X accelerometer errors.**

north and east tilt errors and the filter's estimated tilts with  $+/- 1\sigma$  uncertainty bounds corresponding to those tilts. Shown in Figs. 11.10 and 11.11 are accelerometer biases, estimates, and uncertainty bounds. Tilt errors show less than complete agreement with true tilt errors prior to turntable rotation. Accelerometer bias estimates are poor prior to this time. Subsequent to turntable rotation, tilt and accelerometer bias estimates are improved. These results confirm the



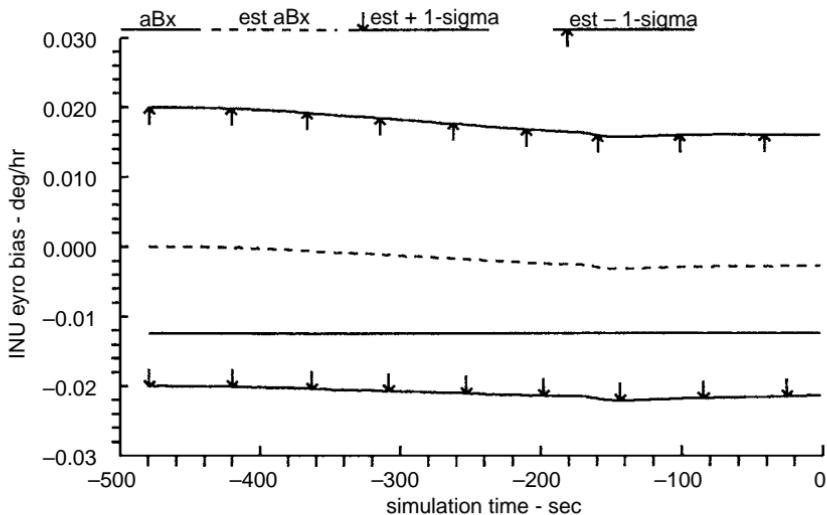
**Fig. 11.11 Alignment Y accelerometer errors.**



**Fig. 11.12** Azimuth simulated alignment.

coupled nature of tilt and accelerometer errors, as predicted by Example 5.1 and earlier error analyses presented in this section. Turntable rotation allows the filter to distinguish between the navigation-frame-referenced tilt and body-referenced accelerometer errors and provides improved estimates of both sources of error as a result.

Azimuth alignment results are shown in Fig. 11.12. This error is slow to improve as a result of this INU's quality. It does converge slowly to the true



**Fig. 11.13** Alignment X gyro errors.

value of azimuth error. For illustration, Fig. 11.13 shows results for one of the gyro bias components. These results demonstrate relatively poor gyro bias estimates achieved over the short alignment time.

## 11.4 Chapter Summary

A method for computing initial attitudes from an inertial navigation system's sensor data was presented. The relationship between initial attitude errors and inertial sensor errors was illustrated.

Simulated INU data were used to illustrate improvements obtained during fine alignment using a Kalman filter implementation. The INU's accelerometer and gyro triad cluster is mounted on a turntable, which rotates during alignment. The fine-alignment filter implementation included accelerometer and gyro sensor error states within its state structure.

These results confirm the coupled nature of tilt and accelerometer errors, as predicted by Example 5.1 and earlier error analyses presented in this section. Turntable rotation allows the filter to distinguish between navigation-frame-referenced tilt and body-referenced accelerometer errors and provide improved estimates of both sources of error as a result. By accomplishing fine alignment, initial attitude errors and inertial sensor errors also can be estimated, yielding improved navigation results.



## Integration via Kalman Filtering: Global-Positioning-System Receiver

In this chapter, the fourth application of the elements from Part I is presented, addressing a navigation system integrating a global-positioning-system (GPS) receiver with an inertial navigation unit (INU). Included in this presentation are GPS receiver Kalman filter configurations, an inertial navigation system (INS) configuration Kalman filter, and a simulated GPS/INS Kalman filter operation.

The Kalman filter implementation described is one of many possible approaches to integrating GPS data with an INU. One approach is to extend the INU tactical-air-navigation (TACAN) integration described in Chapter 10 by using a local-level navigation frame and incorporating satellite ranging measurements to the Kalman filter algorithm. These measurements, being in an Earth-centered, Earth-fixed (ECEF) navigation frame, would require a transformation from the ECEF frame to the navigation frame via this direction cosine matrix in the measurement matrix. This approach would permit additional Kalman filter state vector elements to be included to estimate the INU's inertial sensor errors, for example, accelerometer and gyro.

The implementation presented in this section is intended to interface with a variety of INUs and their associated output variations. Although each INU might output consistent navigation state data (position, velocity, and attitude), the INU's mechanizations can differ, precluding a consistent Kalman filter state vector definition of inertial sensor contributions to the system's error model. Thus, to use the same Kalman filter design for a variety of INUs, it must be generic in the sense that the error dynamic model is limited to navigation state errors that are common to intended INUs.

The GPS receiver's Kalman filter implements two system error models. An INS error model is implemented for operations with INU data, and a position velocity and acceleration (PVA) model is implemented for operations without an INU. Each of these implementations provides refined navigation-frame-referenced position and velocity using estimates by processing raw GPS satellite ranging data (pseudorange and delta range as presented in Chapter 7). The PVA implementation provides refined position, velocity, and time based on a tracking model error dynamic model implemented in the Kalman filter. The INS implementation provides refined navigation state data, position, velocity, and attitude, based on a navigation system error dynamic model implemented in the Kalman filter.

The GPS/inertial integration is illustrated using simulated data to examine characteristics of the GPS receiver's INS Kalman filter.

## 12.1 Global-Positioning-System Receiver Kalman Filter Configurations

### 12.1.1 Position Velocity and Acceleration Configuration Filter

The receiver's PVA-configuration Kalman filter includes 12 states. The states are position, velocity, acceleration, altimeter bias, clock phase and frequency errors. This filter is modeled as a tracking filter (see Sec. 9.7). The PVA configuration's state vector elements are as follows:

$$\begin{aligned}\delta p &= \text{ECEF nominal position error vector } (3 \times 1) \\ \delta v &= \text{ECEF nominal velocity error vector } (3 \times 1) \\ \delta a &= \text{ECEF nominal acceleration error vector } (3 \times 1) \\ \delta h &= \text{baro altimeter bias} \\ b &= \text{range bias (clock phase error} * c; \text{speed of light)} \\ d &= \text{range drift (clock frequency error} * c)\end{aligned}\tag{12.1}$$

### 12.1.2 Inertial-Navigation-System Configuration Filter

The receiver's INS-configuration Kalman filter also includes 12 states. The navigation state errors are position, velocity, and attitude errors referenced in the ECEF frame. Measurement error states of clock phase and frequency are included in the model, whose states are as follows:

$$\begin{aligned}\delta p &= \text{ECEF nominal position error vector } (3 \times 1) \\ \delta v &= \text{ECEF nominal velocity error vector } (3 \times 1) \\ \psi &= \text{ECEF INS platform attitude error vector } (3 \times 1) \\ \delta h &= \text{baro altimeter bias} \\ b &= \text{range bias (clock phase error} * c; \text{speed of light)} \\ d &= \text{range drift (clock frequency error} * c)\end{aligned}\tag{12.2}$$

## 12.2 Inertial-Navigation-System Configuration Kalman Filter

In this section, the navigation-system error model for the GPS receiver's INS configuration Kalman filter will be developed. These developments parallel error equations developed in Chapter 5.

### 12.2.1 Earth-Centered, Earth-Fixed Navigation Solution

The GPS INS filter is implemented in the (ECEF) frame. A position vector in this frame is related to a position vector in an inertial frame by

$$\mathbf{r}^i = C_e^i \mathbf{r}^e \quad (12.3)$$

The first time derivative of this equation is

$$\begin{aligned} \dot{\mathbf{r}}^i &= C_e^i \dot{\mathbf{r}}^e + \dot{C}_e^i \mathbf{r}^e \\ &= C_e^i (\dot{\mathbf{r}}^e + \Omega_{i/e}^e \mathbf{r}^e) \end{aligned} \quad (12.4)$$

The second time derivative of the position vector follows from this equation as

$$\begin{aligned} \ddot{\mathbf{r}}^i &= C_e^i (\ddot{\mathbf{r}}^e + \dot{\Omega}_{i/e}^e \mathbf{r}^e + \Omega_{i/e}^e \dot{\mathbf{r}}^e) + \dot{C}_e^i (\dot{\mathbf{r}}^e + \Omega_{i/e}^e \mathbf{r}^e) \\ &= C_e^i [\ddot{\mathbf{r}}^e + 2\Omega_{i/e}^e \dot{\mathbf{r}}^e + (\dot{\Omega}_{i/e}^e + \Omega_{i/e}^e \Omega_{i/e}^e) \mathbf{r}^e] \end{aligned} \quad (12.5)$$

The specific force plus gravitational is given in terms of the inertial acceleration as

$$C_i^e \ddot{\mathbf{r}}^i = \mathbf{f}^e + \mathbf{G}^e \quad (12.6)$$

Substituting this equation into Eq. (12.5), assuming that  $\dot{\Omega}_{i/e}^e = 0$ , and solving for the ECEF elements yield

$$\ddot{\mathbf{r}}^e + 2\Omega_{i/e}^e \dot{\mathbf{r}}^e + \Omega_{i/e}^e \Omega_{i/e}^e \mathbf{r}^e = \mathbf{f}^e + \mathbf{G}^e \quad (12.7)$$

### 12.2.2 Inertial-Navigation-System Filter Error Dynamic Model

The computed position and rotation rate are assumed to be in error by small quantities such that products of these quantities are negligible. The computed values are represented by the following:

$$\tilde{\mathbf{r}}^e = \mathbf{r}^e + \delta \mathbf{r}^e \quad (12.8)$$

$$\tilde{\Omega}_{i/e}^e = \Omega_{i/e}^e + \delta \Omega_{i/e}^e \quad (12.9)$$

$$\tilde{\mathbf{f}}^e = \mathbf{f}^e + \delta \mathbf{f}^e \quad (12.10)$$

$$\tilde{\mathbf{G}}^e = \mathbf{G}^e + \delta \mathbf{G}^e \quad (12.11)$$

Equation (12.7) is assumed to be true for both the true and computed position. Substituting Eqs. (12.8–12.11) into Eq. (12.7) and solving for the

error variables yield

$$\begin{aligned}\delta\ddot{\mathbf{r}}^e + 2\delta\Omega_{i/e}^e \dot{\mathbf{r}}^e + 2\Omega_{i/e}^e \delta\dot{\mathbf{r}}^e + \Omega_{i/e}^e \Omega_{i/e}^e \delta\mathbf{r}^e + (\delta\Omega_{i/e}^e \Omega_{i/e}^e + \Omega_{i/e}^e \delta\Omega_{i/e}^e) \mathbf{r}^e \\ = \delta\mathbf{f}^e + \delta\mathbf{G}^e\end{aligned}\quad (12.12)$$

Or because it is assumed that  $\delta\Omega_{i/e}^e = 0$ ,

$$\delta\ddot{\mathbf{r}}^e + 2\Omega_{i/e}^e \delta\dot{\mathbf{r}}^e + \Omega_{i/e}^e \Omega_{i/e}^e \delta\mathbf{r}^e = \delta\mathbf{f}^e + \delta\mathbf{G}^e \quad (12.13)$$

The uncertainty in gravitation can be expressed in terms of the ECEF position uncertainty, assuming an ellipsoidal Earth gravity representation without the  $J_2$  term [1], as

$$\delta\mathbf{G}^e = \frac{\mu}{r^3} \left[ 3 \frac{\mathbf{r}^{e^T} \delta\mathbf{r}^e \mathbf{r}^e}{r^2} - \delta\mathbf{r}^e \right] \quad (12.14)$$

Using vector identities, the first term in the brackets can be expressed as

$$\mathbf{r}^{e^T} \delta\mathbf{r}^e \mathbf{r}^e = (\mathbf{r}^e \times) (\mathbf{r}^e \times) \delta\mathbf{r}^e + r^2 \delta\mathbf{r}^e \quad (12.15)$$

where  $(\mathbf{r}^e \times)$  represents the skew-symmetric equivalent of a vector product. Substituting this equation into Eq. (12.14) yields

$$\begin{aligned}\delta\mathbf{G}^e &= \frac{\mu}{r^3} \left[ 3 \frac{(\mathbf{r}^e \times)(\mathbf{r}^e \times)}{r^2} \delta\mathbf{r}^e + 3\delta\mathbf{r}^e - \delta\mathbf{r}^e \right] \\ &= \omega_s^2 \left[ 3 \frac{(\mathbf{r}^e \times)(\mathbf{r}^e \times)}{r^2} + 2I \right] \delta\mathbf{r}^e \\ &= \omega_s^2 [3(\mathbf{e}^e \times)(\mathbf{e}^e \times) + 2I] \delta\mathbf{r}^e \\ &= \omega_s^2 [3(\mathbf{e}^e \mathbf{e}^{e^T} - I) + 2I] \delta\mathbf{r}^e \\ &= \omega_s^2 [3\mathbf{e}^e \mathbf{e}^{e^T} - I] \delta\mathbf{r}^e\end{aligned}\quad (12.16)$$

where  $\mathbf{e}^e$  is the ECEF position unit vector and  $\omega_s$  is the Schuler frequency (see Example 5.1).

The error in specific force includes errors from platform attitude, position error, and accelerometer components. The dependence on attitude and position will be developed in the following sections. Expressing the specific force in the ECEF frame through a series of transformations from the body frame using

transformations defined earlier results in

$$\begin{aligned}
 \bar{\mathbf{f}}^e &= \bar{\mathbf{C}}_n^e \bar{\mathbf{C}}_b^n \bar{\mathbf{f}}^b \\
 &= \mathbf{C}_n^e [\mathbf{I} + (\delta\boldsymbol{\theta} \times)] [\mathbf{I} - (\boldsymbol{\phi} \times)] \mathbf{C}_b^n \bar{\mathbf{f}}^b \\
 &= \mathbf{C}_n^e [\mathbf{I} - (\boldsymbol{\phi} - \delta\boldsymbol{\theta} \times)] \mathbf{C}_b^n \bar{\mathbf{f}}^b \\
 &= \mathbf{C}_n^e [\mathbf{I} - (\boldsymbol{\psi} \times)] \mathbf{C}_b^n \bar{\mathbf{f}}^b \\
 &= \mathbf{C}_n^e [\mathbf{I} - (\boldsymbol{\psi} \times)] \mathbf{C}_e^n \mathbf{C}_n^e \mathbf{C}_b^n \bar{\mathbf{f}}^b \\
 &= [\mathbf{I} - \mathbf{C}_n^e (\boldsymbol{\psi} \times) \mathbf{C}_e^n] \mathbf{C}_n^e \mathbf{C}_b^n \bar{\mathbf{f}}^b
 \end{aligned} \tag{12.17}$$

Redefine the preceding  $\boldsymbol{\psi}$  attitude variable using the following similarity transformation:

$$(\boldsymbol{\Psi} \times) \equiv \mathbf{C}_n^e (\boldsymbol{\psi} \times) \mathbf{C}_e^n \tag{12.18}$$

Then, the position and attitude error contributions to the specific force error can be expressed as

$$\begin{aligned}
 \delta\mathbf{f}^e &= -(\boldsymbol{\Psi} \times) \mathbf{f}^e \\
 &= \mathbf{f}^e \times \boldsymbol{\Psi}
 \end{aligned} \tag{12.19}$$

Substituting Eq. (12.16) and Eq. (12.19) into Eq. (12.13) yields

$$\delta\dot{\mathbf{r}}^e = -\omega_s^2 (\mathbf{I} - 3\mathbf{e}^e \mathbf{e}^{e^T}) \delta\mathbf{r}^e - \Omega_{i/e}^e \Omega_{i/e}^e \delta\mathbf{r}^e - 2\Omega_{i/e}^e \delta\dot{\mathbf{r}}^e + \mathbf{f}^e \times \boldsymbol{\Psi} \tag{12.20}$$

The second position error term containing Earth rotation rates is small compared to the first term containing the Schuler frequency and is not included.

The INS Kalman filter is intended to be generic and operate with many different INUs. As such, its Kalman filter system error dynamic model does not include additional error states that would represent INU instrument errors and accelerometer and gyro biases. To some extent, the absence of these error states in the Kalman filter is accounted for by the use of process noise. As indicated by Table 12.1, this process noise is specified separately for horizontal and vertical influences. The process noise matrix will be developed in the following.

The Kalman filter error covariance matrix is propagated by the following continuous matrix equivalent of Eq. (8.42):

$$P(t) = \Phi(t, t_0) P(t_0) \Phi(t, t_0)^T + \int_{t_0}^t \Phi(t, \tau) Q(\tau) \Phi(t, \tau)^T d\tau \tag{12.21}$$

The first term is computed as is; however, the second term is implemented in its completed analytical form. This second term is evaluated in the following.

**Table 12.1 GPS receiver INS configuration tuning parameters [20]**

Parameter		Description	Value
$N_{hp}$ ,	$\text{m}^2/\text{s}$	Horizontal position plant noise	1
$N_{vp}$ ,	$\text{m}^2/\text{s}$	Vertical position plant noise	1
$N_{hv}$ ,	$\text{m}^2/\text{s}^3$	Horizontal velocity plant noise	$1.\text{E}-2^{\text{a}}$
$N_{vv}$ ,	$\text{m}^2/\text{s}^3$	Vertical velocity plant noise	$1.\text{E}-2$
$N_f$ ,	$1/\text{s}$	Specific force-dependent plant noise	$1.\text{E}-6$
$N_{\psi p}$ ,	$\text{rad}^2/\text{s}$	Misalignment $\psi$ plant noise	$1.\text{E}-12$
$N_{\text{alt}:a_1}$ ,	$\text{m}^2/\text{s}$	Baro altimeter bias plant noise	$1.\text{E}+1$
$N_{\text{alt}:a_2}$ ,	s	Velocity-dependent baro altimeter plant noise	$1.\text{E}-3$
$N_b$ ,	$\text{m}^2/\text{s}$	Clock plant noise	0.25
$N_d:d_1$ ,	$\text{m}^2/\text{s}^3$	Clock rate plant noise	$2.\text{E}-4$
$N_d:d_2$ ,	s	Acceleration-dependent clock rate plant noise	$1.\text{E}-4$
$R_{\text{pseudo}}$ ,	$\text{m}^2$	Pseudorange measurement noise	2.25
$R_{\text{delta}}$ ,	$\text{m}^2$	Delta-range measurement noise	$95.\text{E}-4$
$k_1$ ,	$\text{I}/\text{s}$	Baro inertial coupling constant	$1.\text{E}-4$
$k_2$ ,	$\text{I}/\text{s}^2$	Baro inertial coupling constant	$1.\text{E}-6$
$\tau_h$ ,	s	Baro altimeter bias correlation time	$1.\text{E}+3$

<sup>a</sup>This value is changed to be consistent with other documentation.

The state transition matrix is approximated by

$$\Phi(t,t_0) \approx I + F\Delta t + F^2 \frac{\Delta t^2}{2!} \quad (12.22)$$

To compute the process noise integral, the dynamics matrix for the INS Kalman filter is approximated by a few nonzero partitions as

$$F(t) \approx \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & (f^e \times) & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & F_{4,4} \end{bmatrix} \quad (12.23)$$

where

$$(f^e \times) = \begin{bmatrix} 0 & -f_3 & f_2 \\ f_3 & 0 & -f_1 \\ -f_2 & f_1 & 0 \end{bmatrix} \quad (12.24)$$

and

$$\Delta t = t - t_0 \quad (12.25)$$

It is assumed that elements in the state transition matrix, that is, specific forces, are time invariant over the integration interval.

The process noise matrix is established as a diagonal matrix, with  $3 \times 3$  diagonal matrix partitions along the diagonal representing process noise for position, velocity, tilt, altimeter, and clock states. Position and velocity  $3 \times 3$  matrix partitions are established to allow different levels of process noise for local-level navigation frame horizontal and vertical terms to be specified. This will be addressed later. Consider the process noise matrix

$$Q(t) \equiv \begin{bmatrix} Q'_{1,1} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & Q''_{2,2} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & Q_{3,3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & Q_{4,4} \end{bmatrix} \quad (12.26)$$

The product inside the integral in Eq. (12.21) becomes

$$\begin{aligned} \Phi Q \Phi^T &= \begin{bmatrix} I & I\Delta t & (f \times) \frac{\Delta t^2}{2} & 0 \\ 0 & I & (f \times) \Delta t & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & \Phi_{4,4} \end{bmatrix} \begin{bmatrix} Q'_{1,1} & 0 & 0 & 0 \\ 0 & Q''_{2,2} & 0 & 0 \\ 0 & 0 & Q_{3,3} & 0 \\ 0 & 0 & 0 & Q_{4,4} \end{bmatrix} \Phi^T \\ &= \begin{bmatrix} Q'_{1,1} + Q''_{2,2} \Delta t^2 + (f \times) Q_{3,3} (f \times)^T \frac{\Delta t^4}{4} & Q''_{2,2} \Delta t + (f \times) Q_{3,3} (f \times)^T \frac{\Delta t^3}{2} \\ [sym] & Q''_{2,2} + (f \times) Q_{3,3} (f \times)^T \Delta t^2 \\ [sym] & [sym] \\ [sym] & [sym] \end{bmatrix} \\ &\quad \begin{bmatrix} (f \times) Q_{3,3} \frac{\Delta t^2}{2} & 0 \\ (f \times) Q_{3,3} \Delta t & 0 \\ Q_{3,3} & 0 \\ [sym] & \Phi_{4,4} Q_{4,4} \Phi_{4,4}^T \end{bmatrix} \quad (12.27) \end{aligned}$$

The  $Q''_{2,2}$  and  $Q_{3,3}$  matrix partitions are defined as

$$Q''_{2,2} = Q'_{2,2} + \Delta t^2 |\mathbf{a}|^2 N_f I \quad (12.28)$$

$$Q_{3,3} = N_\psi I \quad (12.29)$$

where  $|\mathbf{a}|$  is the magnitude of the acceleration vector  $\mathbf{a}$ .

Substituting these definitions into Eq. (12.27) and integrating over the interval yield the following for each of the partitions:

1,3:

$$\int (\mathbf{f} \times) Q_{3,3} \frac{\Delta t^2}{2} d\tau = (\mathbf{f} \times) N_\psi \frac{\Delta t^3}{6} \quad (12.30)$$

2,3:

$$\int (\mathbf{f} \times) Q_{3,3} \Delta t d\tau = (\mathbf{f} \times) N_\psi \frac{\Delta t^2}{2} \quad (12.31)$$

3,3:

$$\int Q_{3,3} d\tau = N_\psi I \Delta t \quad (12.32)$$

1,2:

$$\begin{aligned} & \int \left[ (Q'_{2,2} + |\mathbf{a}|^2 N_f I \Delta t^2) \Delta t + (\mathbf{f} \times) Q_{3,3} (\mathbf{f} \times)^T \frac{\Delta t^2}{2} \right] d\tau \\ &= Q'_{2,2} \frac{\Delta t^2}{2} + |\mathbf{a}|^2 N_f I \frac{\Delta t^4}{4} + (\mathbf{f} \times) (\mathbf{f} \times)^T N_\psi \frac{\Delta t^4}{8} \end{aligned} \quad (12.33)$$

2,2:

$$\begin{aligned} & \int \left[ (Q'_{2,2} + |\mathbf{a}|^2 N_f I \Delta t^2) + (\mathbf{f} \times) Q_{3,3} (\mathbf{f} \times)^T \Delta t^2 \right] d\tau \\ &= Q'_{2,2} \Delta t + |\mathbf{a}|^2 N_f I \frac{\Delta t^3}{3} + (\mathbf{f} \times) (\mathbf{f} \times)^T N_\psi \frac{\Delta t^3}{3} \end{aligned} \quad (12.34)$$

1,1:

$$\begin{aligned} & \int \left[ Q'_{1,1} + (Q'_{2,2} + |\mathbf{a}|^2 N_f I \Delta t^2) \Delta t^2 + (\mathbf{f} \times) Q_{3,3} (\mathbf{f} \times)^T \frac{\Delta t^4}{4} \right] d\tau \\ &= Q'_{1,1} \Delta t + Q'_{2,2} \frac{\Delta t^3}{3} + |\mathbf{a}|^2 N_f I \frac{\Delta t^5}{5} + (\mathbf{f} \times) (\mathbf{f} \times)^T N_\psi \frac{\Delta t^5}{20} \end{aligned} \quad (12.35)$$

It is desirable to express the preceding  $Q'_{1,1}$  and  $Q'_{2,2}$  matrix partitions such that the local-level navigation-frame horizontal and vertical process noise levels can be specified separately and directly as

$$Q'_{1,1} = C_L^e Q_{1,1} (C_L^e)^T \quad (12.36)$$

and

$$Q'_{2,2} = C_L^e Q_{2,2} (C_L^e)^T \quad (12.37)$$

Expanding these equations yields

$$\begin{aligned} Q' &= C_L^e Q (C_L^e)^T \\ &= \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} N_h & 0 & 0 \\ 0 & N_h & 0 \\ 0 & 0 & N_v \end{bmatrix} (C_L^e)^T \\ &= \begin{bmatrix} (C_{11}C_{11} + C_{12}C_{12})N_h + C_{13}C_{13}N_v & (C_{11}C_{21} + C_{12}C_{22})N_h + C_{13}C_{23}N_v & \dots \\ (C_{21}C_{11} + C_{22}C_{12})N_h + C_{23}C_{13}N_v & (C_{21}C_{21} + C_{22}C_{22})N_h + C_{23}C_{23}N_v & \dots \\ (C_{31}C_{11} + C_{32}C_{12})N_h + C_{33}C_{13}N_v & (C_{31}C_{21} + C_{32}C_{22})N_h + C_{33}C_{23}N_v & \dots \end{bmatrix} \end{aligned} \quad (12.38)$$

This trend is repeated for the remaining column. If the diagonal matrix containing the  $N$  just given is replaced with the identity matrix, then the component of this product is the identity matrix

$$C_L^e (C_L^e)^T = \begin{bmatrix} C_{11}C_{11} + C_{12}C_{12} + C_{13}C_{13} & C_{11}C_{21} + C_{12}C_{22} + C_{13}C_{23} & \dots \\ C_{21}C_{11} + C_{22}C_{12} + C_{23}C_{13} & C_{21}C_{21} + C_{22}C_{22} + C_{23}C_{23} & \dots \\ C_{31}C_{11} + C_{32}C_{12} + C_{33}C_{13} & C_{31}C_{21} + C_{32}C_{22} + C_{33}C_{23} & \dots \end{bmatrix} \quad (12.39)$$

The last column of the  $C_L^e$  matrix is the unit vector of the ECEF position. Its outer product, in terms of the matrix elements just defined, is

$$\begin{aligned} \mathbf{e} \mathbf{e}^T &= \begin{bmatrix} C_{13} \\ C_{23} \\ C_{33} \end{bmatrix} [C_{13} \ C_{23} \ C_{33}] \\ &= \begin{bmatrix} C_{13}C_{13} & C_{13}C_{23} & C_{13}C_{33} \\ C_{23}C_{13} & C_{23}C_{23} & C_{23}C_{33} \\ C_{33}C_{13} & C_{33}C_{23} & C_{33}C_{33} \end{bmatrix} \end{aligned} \quad (12.40)$$

Equation (12.38) can be obtained by subtracting this equation from Eq. (12.39), multiplying the result by  $N_h$ , and adding this equation, multiplied by  $N_v$ . Or

$$C_L^e Q(C_L^e)^T = N_h(I - \mathbf{e} \mathbf{e}^T) + N_v \mathbf{e} \mathbf{e}^T \quad (12.41)$$

The outer product of the position unit vector is defined as

$$\Pi \equiv \mathbf{e} \mathbf{e}^T \quad (12.42)$$

With this definition, Eq. (12.41) is rewritten as

$$\begin{aligned} C_L^e Q(C_L^e)^T &= N_h(I - \Pi) + N_v \Pi \\ &= N_h I + (N_v - N_h) \Pi \end{aligned} \quad (12.43)$$

This expression is substituted for the  $Q'$  matrices in Eqs. (12.33–12.35), thus establishing a means to “tune” the filter with local-level navigation-frame specified maneuvering characteristics.

*Dynamics Matrix:*

$$F(t) = \begin{bmatrix} -k_1 \Pi & I_{3 \times 3} & 0_{3 \times 3} & [k_1 \mathbf{e} \quad 0_{3 \times 2}] \\ -\omega_s^2 I_{3 \times 3} - (k_2 - 3\omega_s^2) \Pi - 2(\boldsymbol{\omega}_{i/e}^e \times) & (\boldsymbol{f}^e \times) & [k_2 \mathbf{e} \quad 0_{3 \times 2}] \\ 0_{3 \times 3} & 0_{3 \times 3} & -(\boldsymbol{\omega}_{i/e}^e \times) & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & F_{4,4} \end{bmatrix}$$

where

$$F_{4,4} = \begin{bmatrix} -1/\tau_h & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

and

- $k_1$  - INS vertical velocity damping coefficient
- $k_2$  - INS vertical acceleration damping coefficient
- $\tau_h$  - baro altimeter bias correlation time

*Noise Distribution Matrix:*

$$G(t) = \begin{bmatrix} C_L^e & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & C_L^e & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & G_{4,4} \end{bmatrix}$$

where

$$G_{4,4} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

**Fig. 12.1 GPS receiver INS Kalman filter system error model.**

A final refinement to Eqs. (12.33–12.35) is obtained by reexpressing the product of the skew-symmetric matrices  $(f \times)(f \times)^T$ . This outer product can be expressed as

$$(f \times)(f \times)^T = |f|^2 - ff^T \quad (12.44)$$

### 12.2.3 Inertial-Navigation-System Filter Error Dynamic Model Summary

The system error dynamic model is summarized in Fig. 12.1. This model is based on an INU system error model formulated in the ECEF frame. Tuning parameters for the filter are defined in the local-level navigation frame and are listed in Table 12.1. Consequently, the noise distribution matrix  $G$  transforms the local-level defined tuning parameters into the ECEF frame.

The corresponding measurement model summary is presented in Fig. 12.2.

## 12.3 Simulated Global-Positioning-System-Receiver Inertial-Navigation-System Kalman Filter Operation

In this operation, simulated data are used to illustrate the GPS receiver's internal Kalman filter operation. The simulation data flow is presented in Fig. 12.3. In this simulation, an environment model generates aircraft accelerations and rates and satellite range and range rates based on a specified aircraft flight profile. Simulated aircraft accelerations and rates plus initialization errors and INU sensor errors are supplied to a dynamic INU model, implementing

*Pseudo Range:*

$$\rho = |\Delta r| + b$$

$$H_{\text{pseudo}} = \begin{bmatrix} \mathbf{e}_{LOS}^{eT} & 0_{1 \times 3} & 0_{1 \times 3} & 0 & 1 & 0 \end{bmatrix}$$

*Delta Range/Pseudo Range Rate:*

$$\Delta\rho = r_f - r_b + d\Delta t$$

$$H_{\text{delta}} = \begin{bmatrix} 0_{1 \times 3} & \mathbf{e}_{LOS}^{eT} \Delta t & 0_{1 \times 3} & 0 & 0 & \Delta t \end{bmatrix}$$

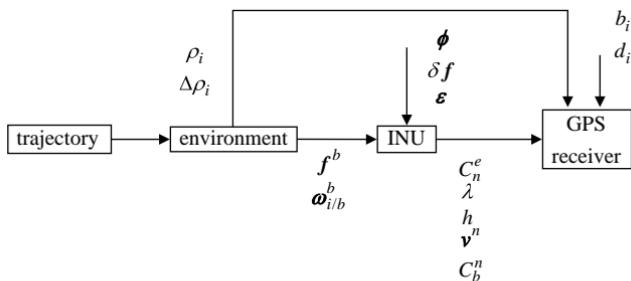
where

$r$  - range from antenna to satellite

$\mathbf{e}_{LOS}^e$  - unit line-of-sight vector from antenna to satellite

$\Delta t$  - difference between final time  $t_f$  and beginning time  $t_b$  for Doppler integration interval.

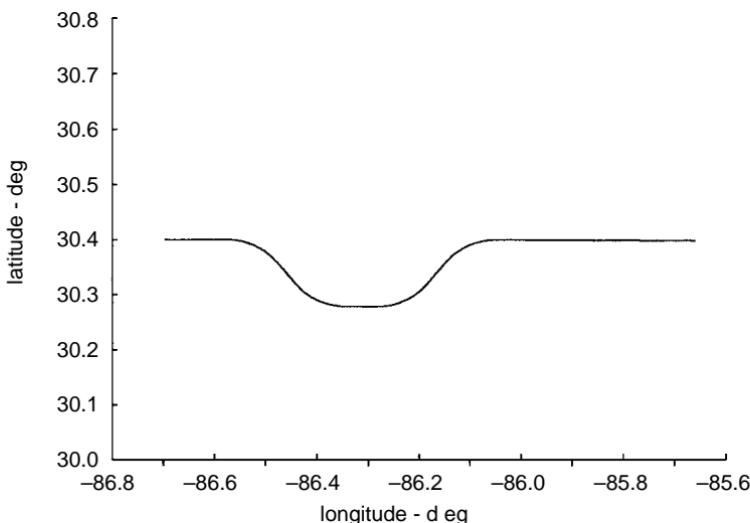
**Fig. 12.2 GPS receiver measurement model.**



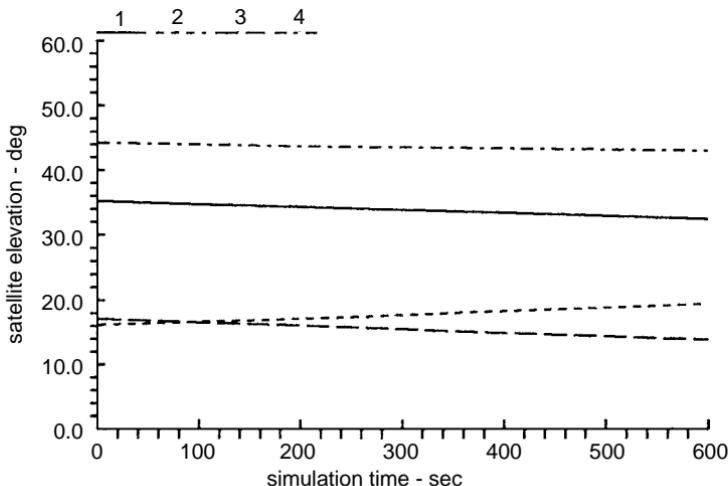
**Fig. 12.3 GPS receiver simulation function flow.**

nonlinear navigation equations presented in Sec. 5.2. Simulated satellite range and range rates are corrupted with bias and noise and are supplied to the GPS receiver model. Navigation state data from the INU model are sent to the GPS-receiver model and converted to ECEF using equations developed in Sec. 4.1 prior to their use in the receiver's Kalman filter. After ECEF frame state estimates in Eq. (12.2) are computed, the reverse of this conversion is used to reestablish corrected data in a local-level reference frame.

This case study is used to assess the performance of the GPS receiver's Kalman filter during a slight horizontal plane maneuver with initial INU attitude errors. The objective of the simulation is to evaluate the filter's estimates for these errors. The horizontal plane maneuver is presented in Fig. 12.4. Four simulated GPS satellites' elevation angles and resulting geometrical



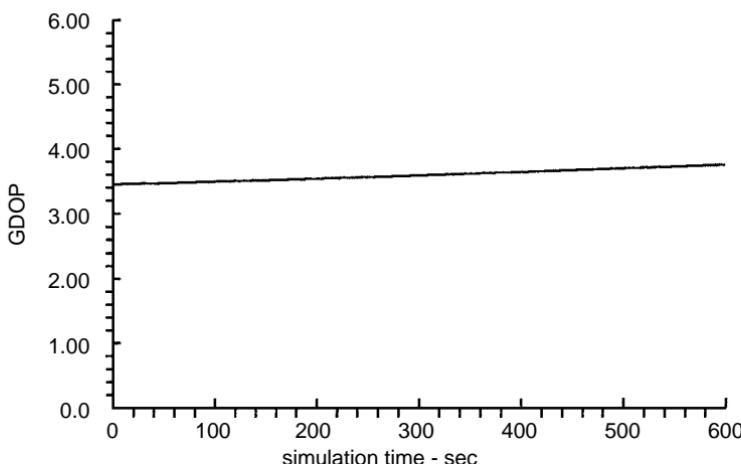
**Fig. 12.4 Simulated trajectory for GPS receiver INS Kalman filter.**



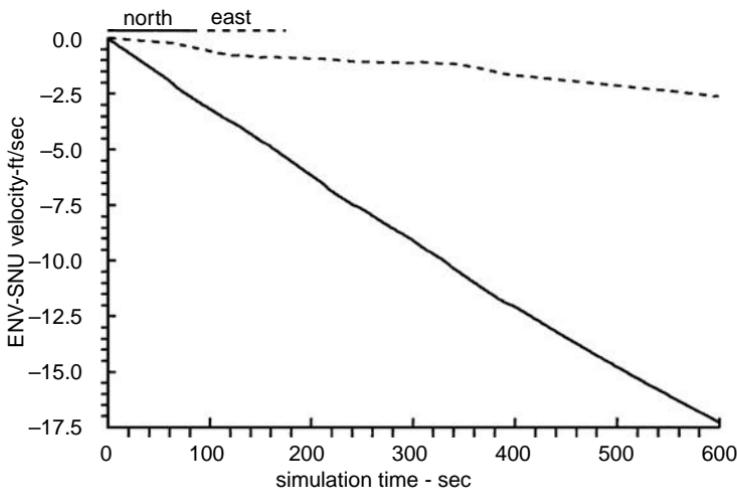
**Fig. 12.5** Simulated GPS satellites' elevations.

dilution of precision (GDOP) (see Sec. 7.3) are presented in Figs. 12.5 and 12.6, respectively.

Simulated INU north and east velocity errors resulting from initial attitude errors are shown in Fig. 12.7. Slight transients in velocity components are seen that are the result of the maneuver shown in Fig. 12.4. Shown in Fig. 12.8 are the differences between the environment and INU velocity difference (the navigation system's true velocity error) and the filter's estimate of that error. This difference should be, and appears to be, zero mean.



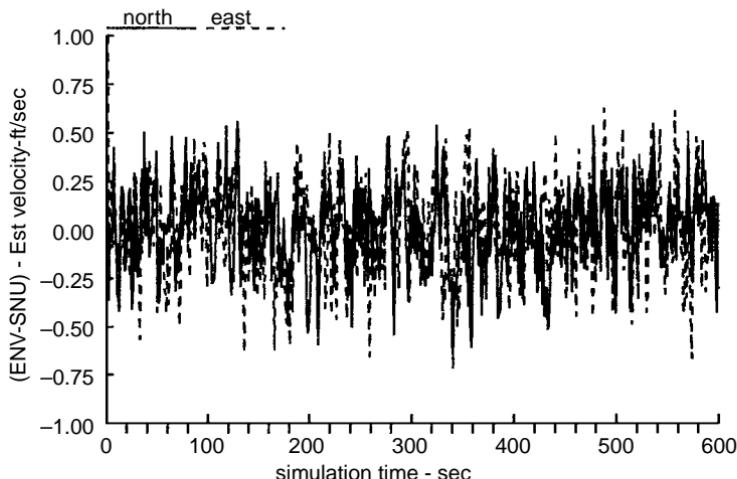
**Fig. 12.6** GDOP for simulated satellites.



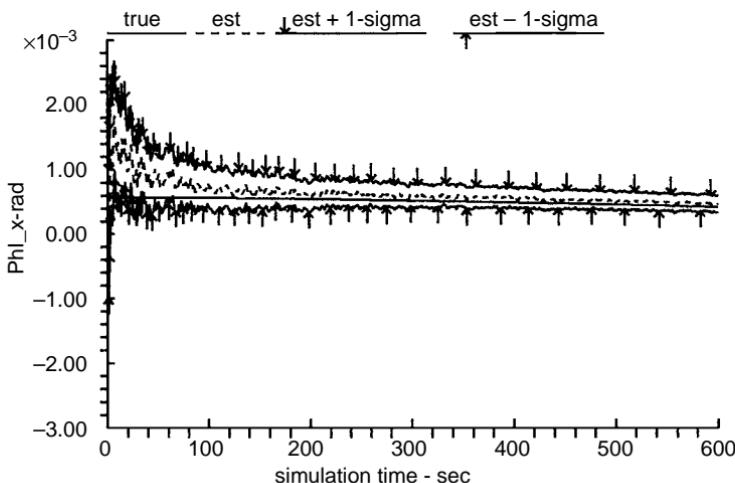
**Fig. 12.7** Simulated INU's velocity errors.

Recall from the derivation of the GPS Kalman filter model that filter states include the  $\psi$  attitude error.  $\phi$  attitude errors in the local-level wander-azimuth frame are of interest and are presented in Figs. 12.9 and 12.10 for the wander-azimuth frame  $x$  and  $y$  components, respectively. These results show that the filter estimates agree with the true tilt errors simulated.

Finally, azimuth errors are presented in Fig. 12.11. The filter does not estimate this error as accurately as horizontal tilt errors presented earlier. Only during the

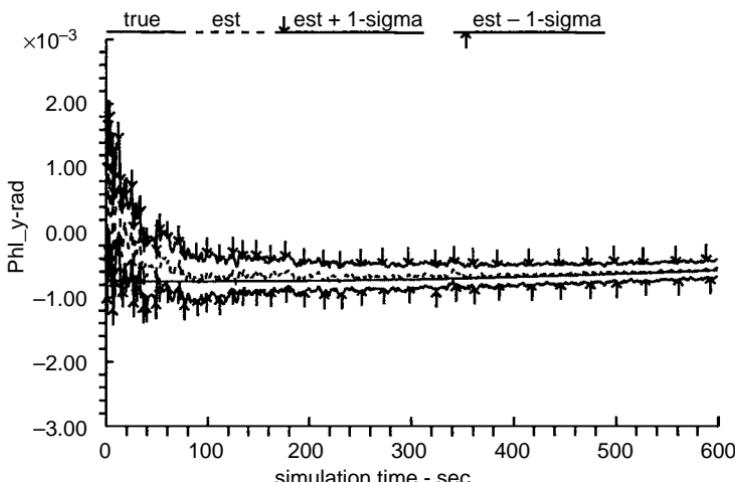


**Fig. 12.8** GPS receiver INS Kalman filter velocity estimation error.

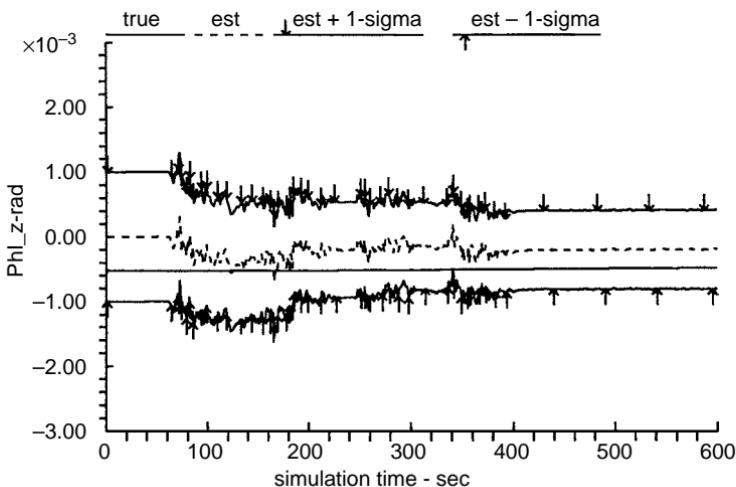


**Fig. 12.9 GPS receiver INS Kalman filter x-tilt error estimate.**

maneuver does the filter improve the estimate. The relatively poor performance of the filter in estimating this error is the result of using tuning parameter values listed in Table 12.1. Smaller values for the horizontal plant noise parameter would result in improved estimates for this error for the INU and its error characteristics simulated. However, the use of a smaller tuning parameter value must be weighted against possible incompatibility with other INUs and their error characteristics.



**Fig. 12.10 GPS receiver INS Kalman filter y-tilt error estimate.**



**Fig. 12.11 GPS receiver INS Kalman filter azimuth error estimate.**

## 12.4 Chapter Summary

The Kalman filter implementation described is one of many possible approaches to integrating GPS data with an INU. This implementation is intended to interface with a variety of INUs and is designed to be generic in the sense that the error dynamic model is limited to navigation state errors that are common among most INUs.

GPS receiver PVA and INS configuration Kalman filters were described. Each of these configurations provides refined navigation-frame-referenced position and velocity using estimates from an internal Kalman filter by processing raw GPS satellite ranging data (pseudorange and delta range as presented in Chapter 7).

The GPS receiver's generic INS Kalman filter form was derived. Details of process noise terms are presented as a result of their increased importance because the interfacing navigation systems' sensor characteristics are unknown and are not included in the filter's state structure.

GPS/inertial integration was illustrated using simulated data to examine characteristics of the GPS receiver's INS Kalman filter. The ability of the INS Kalman filter to estimate known simulated errors was demonstrated. The quality of these estimates is based on tuning parameter values for this implementation.

## In-Motion Alignment

In this chapter, the fifth application of the elements in Part 1, two subproblems of inertial measurement unit (IMU) in-motion alignment, are presented: transfer-alignment and alignment without benefit of attitude initialization. The problem of in-motion alignment differs from that presented for ground alignment in Chapter 11 in that the Earth's gravity and rotation inherent characteristics cannot be used for alignment. To accomplish in-motion alignment, other sensors and/or assumptions must be employed. Common to both in-motion and ground alignments, the principal problem is initialization of the navigation state's differential equations.

If, for example, a magnetic compass' output were available, then it is a source of heading, although imprecise because of local and magnetic field deviations, which can be used to initialize the attitude equations. If the vehicle is an aircraft and is flying wings roughly level, then roll and pitch angles can be assumed to be zero. With the combinations of the compass' heading and zero roll and pitch,  $C_b^w$  matrix elements can be initialized. Initialization of other navigation states, for example, position and velocity, can be accomplished with an external source, for example, global positioning system (GPS), tactical air navigation (TACAN) plus output from an airspeed sensor, etc. These same external sources of position and/or velocity can then be used to refine the alignment via an alignment Kalman filter. Reasonably accurate initializations permit the use of error dynamics models based on small attitude errors, as presented in Chapter 5.

If attitude data are not available for initialization, then the alignment Kalman filter error model is reformulated for large attitude errors. This large attitude error form is used for the initial phases of alignment until the attitude errors are reduced to levels consistent with small attitude error model assumptions.

Tactical quality IMU's test data are used to illustrate the two types of in-motion alignment. Because an IMU is used, a navigation solution for position, velocity, and attitude must be generated from the IMU's  $\Delta v$  and  $\Delta\theta$  outputs. A navigation solution is generated from position, velocity, and attitude differential equations presented in Chapter 5. Initialization of these equations and associated navigation system error modeling is presented for both small and large heading errors.

### 13.1 Transfer Alignment

#### 13.1.1 Transfer-Alignment Process

A carriage aircraft's navigation states are the data sources for initializing navigation differential equations implemented in a local-level wander-azimuth navigation frame.

Once initialized, navigation states (position, velocity, and attitude) are obtained by numerically integrating corresponding differential equations using as inputs the IMU's  $\Delta v$  and  $\Delta\theta$  outputs. Using the aircraft's navigation states as initial conditions for the IMU's navigation solution assumes that the aircraft and IMU body-axis alignments are approximately known with only small misalignments between them, that is, structure misalignments. This assumption allows the use of a small attitude error model for navigation system error equations.

It is the task of the transfer-alignment process to estimate initialization errors and errors associated with the IMU's instruments, that is, accelerometers and gyros. The aircraft's navigation state positions are processed as observations in the transfer-alignment Kalman filter algorithm. Error estimates from the alignment filter for IMU's navigation states are used to correct, closed-loop, the ongoing navigation solution. Data flow is shown in Fig. 13.1.

#### 13.1.2 Navigation Equations

Navigation equations are implemented in a local-level wander-azimuth frame. The relationship between the geographic frame's *N-E-D* axes and the wander-azimuth frame's *X-Y-Z* axes was presented in Sec. 3.1.

*Position* is maintained in the navigation-to-Earth direction cosine matrix. This matrix, for the definition of the wander-azimuth angle  $\alpha$  given

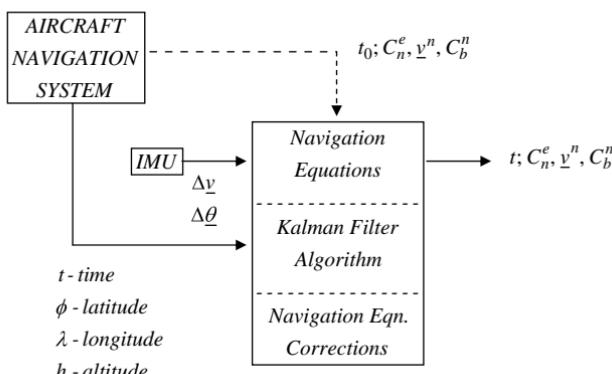


Fig. 13.1 Transfer-alignment data flow.

in Fig. 3.4, is

$$C_n^e(z) = \begin{bmatrix} -cas\phi c\lambda - sas\lambda & sas\phi c\lambda - cas\lambda & -c\phi c\lambda \\ -cas\phi s\lambda + s\alpha c\lambda & sas\phi s\lambda + c\alpha c\lambda & -c\phi s\lambda \\ c\alpha c\phi & -s\alpha c\phi & -s\phi \end{bmatrix} \quad (13.1)$$

where  $s$  and  $c$  indicate sine and cosine functions, respectively. The evolution of this matrix is governed by the following matrix differential equation:

$$\dot{C}_e^n = -\Omega_{e/n}^n C_e^n \quad (13.2)$$

where  $\Omega_{e/n}^n$  is the skew-symmetric matrix equivalent of the vector cross product ( $\omega_{e/n}^n \times$ ), and this vector is the rotation rate vector from the navigation frame with respect to the Earth-centered, Earth-fixed (ECEF) frame, referenced in the navigation frame, and is also known as the transport rate. This transport rate vector  $\omega_{e/n}^n$  is expressed in terms of the wander-azimuth frame velocities:

$$\omega_{e/n}^n \equiv \boldsymbol{\rho} \approx \begin{bmatrix} \frac{v_y}{R_y} \\ -\frac{v_x}{R_x} \\ \rho_z \end{bmatrix} \quad (13.3)$$

The wander-azimuth implementation specifies

$$\rho_z \equiv 0 \quad (13.4)$$

*Velocity* is obtained from the vector differential equation

$$\dot{\mathbf{v}}^n = \mathbf{f}^n - (\Omega_{e/n}^n + 2\Omega_{i/e}^n)\mathbf{v}^n + \mathbf{g}^n \quad (13.5)$$

where

$\mathbf{f}^n$  = specific force vector in navigation frame (transformed from body-referenced IMU's  $\Delta v$  outputs)

$\Omega_{i/e}^n$  = skew-symmetric matrix equivalent of the vector cross product ( $\omega_{i/e}^n \times$ ) where the rotation rate vector is from the ECEF frame with respect to an inertial frame referenced in the navigation frame

$\mathbf{g}^n$  = gravity vector in navigation frame

The Earth rotation vector is expressed in wander-azimuth-frame components as

$$\boldsymbol{\omega}_{i/e}^n \equiv \boldsymbol{\Omega} = C_e^n \begin{bmatrix} 0 \\ 0 \\ \omega_{i/e} \end{bmatrix} \quad (13.6)$$

*Attitude* is maintained in the body-to-navigation frame direction cosine matrix. This matrix is given as

$$C_b^n(z) = \begin{bmatrix} c\theta c\psi_{az} & -c\varphi s\psi_{az} + s\varphi s\theta c\psi_{az} & s\varphi s\psi_{az} + c\varphi s\theta c\psi_{az} \\ c\theta s\psi_{az} & c\varphi c\psi_{az} + s\varphi s\theta s\psi_{az} & -s\varphi c\psi_{az} + c\varphi s\theta s\psi_{az} \\ -s\theta & s\varphi c\theta & c\varphi c\theta \end{bmatrix} \quad (13.7)$$

where  $\psi_{az}$  is related to the wander-azimuth angle and true heading by

$$\psi_T = \alpha + \psi_{az} \quad (13.8)$$

The evolution of the  $C_b^n$  matrix is governed by the following matrix differential equation:

$$\dot{C}_n^b = -\boldsymbol{\Omega}_{n/b}^b C_n^b \quad (13.9)$$

where, using the angular velocity addition theorem, the angular rotation matrix  $\boldsymbol{\Omega}_{n/b}^b$  is obtained from the elements of the following angle rate vector:

$$\boldsymbol{\omega}_{n/b}^b = \boldsymbol{\omega}_{l/b}^b - C_n^b (\boldsymbol{\omega}_{e/n}^n + \boldsymbol{\omega}_{i/e}^n) \quad (13.10)$$

The first term in this equation represents the IMU's  $\Delta\theta$  outputs.

The IMU's  $\Delta v$  and  $\Delta\theta$  outputs are available at high rate. These outputs are summed down to a lower rate for numerical integration of navigation differential equations. Kalman filter estimates are applied as corrections after being reformulated into a more convenient form. Available out of the navigation equation (Kalman filter algorithm) correction block in Fig. 13.1 are corrected navigation state data.

### 13.1.3 Navigation System Error Equations

Navigation system error equations presented in this section are based on the velocity error representation for the computer frame. Position and attitude error equations will be followed by velocity error equations.

*Position.* Expressing computed values with an overbar, the computed position direction cosine matrix is represented as

$$\tilde{C}_e^n = [I - (\delta\boldsymbol{\theta}_x)] C_e^n \quad (13.11)$$

The following position error vector differential equation was established in Sec. 5.3:

$$\delta \dot{\theta} = \delta \omega_{e/n}^n - \omega_{e/n}^n \times \delta \theta \quad (13.12)$$

It is assumed that  $\delta \theta_z = 0$ , which implies  $\delta \dot{\theta}_z = 0$  expanding the components in this equation, the following expression is obtained from the third element:

$$\delta \rho_z = -\rho_y \delta \theta_x + \rho_x \delta \theta_y \quad (13.13)$$

*Attitude.* The computed attitude direction cosine matrix is represented as

$$\bar{C}_b^n = [I - (\phi x)] C_b^n \quad (13.14)$$

The following vector differential equation for the attitude error was presented in Sec. 5.3:

$$\dot{\phi} = \delta \omega_{e/n}^n + \omega_{i/e}^n \times \delta \theta + \phi \times (\omega_{i/e}^n + \omega_{e/n}^n) + \epsilon^n \quad (13.15)$$

*Velocity.* The computed velocity vector is represented by the following error representation for the computer frame velocity error (denoted by the superscript 1):

$$\bar{v}^n = [I - (\delta \theta \times)] v^n + \delta v^1 \quad (13.16)$$

This form results in the following vector differential equation for the velocity error:

$$\begin{aligned} \delta \dot{v}^1 &= (\delta \theta - \phi) \times f^n - (\omega_{e/n}^n + 2\omega_{i/e}^n) \times \delta v^1 \\ &\quad + \delta f^n + \delta g^n - g^n \times \delta \theta \end{aligned} \quad (13.17)$$

The velocity error definition in Eq. (13.16) yields for the transport rate error vector in Eq. (13.15)

$$\delta \omega_{e/n}^n \equiv \delta \rho = \begin{bmatrix} \frac{v_z}{R} \delta \theta_x + \frac{1}{R} \delta v_y^1 - \frac{v_x}{R} \delta \theta_z - \frac{v_y}{R^2} \delta h \\ \frac{v_z}{R} \delta \theta_y - \frac{1}{R} \delta v_x^1 - \frac{v_y}{R} \delta \theta_z + \frac{v_x}{R^2} \delta h \\ \delta \rho_z \end{bmatrix} \quad (13.18)$$

The tilt attitude error defined in Eq. (13.14) is related to the angular position error defined in Eq. (13.11) and the  $\psi$  attitude by

$$\phi = \psi + \delta \theta \quad (13.19)$$

Because  $\delta \theta_z \equiv 0$ , then  $\psi_z = \phi_z$ . The third attitude error state is the  $\psi$  variable  $\psi_z$ . The preceding error dynamic equations are summarized in Fig 13.2.

$$\begin{bmatrix}
 \frac{v_z}{R} & 0 & -\frac{v_y}{R^2} & 0 & \frac{1}{R} & 0 & 0 & 0 & 0 \\
 0 & \frac{v_z}{R} & \frac{v_x}{R} & -\frac{1}{R} & 0 & 0 & 0 & 0 & 0 \\
 \delta h_y & v_y & R & 0 & 0 & -1 & 0 & 0 & 0 \\
 \delta h & 0 & -v_x & 0 & 0 & (\rho + 2\Omega)_y & 0 & -f_z & f_y \\
 \delta v_x^1 & 0 & (f_z + g) & 0 & 0 & 2\Omega_z & -f_z & 0 & -f_x \\
 \frac{d}{dt} \delta v_y^1 & -(f_z + g) & 0 & 0 & -2\Omega_z & 0 & f_z & 0 & -f_x \\
 \delta v_z^1 & f_y & -f_x & -2\frac{g}{R} & (\rho + 2\Omega)_x & 0 & -f_y & f_x & 0 \\
 \phi_x & \frac{v_z}{R} & -\Omega_z & -\frac{v_y}{R^2} & 0 & \frac{1}{R} & 0 & 0 & \Omega_z \\
 \phi_y & \Omega_z & \frac{v_z}{R} & \frac{v_x}{R^2} & -\frac{1}{R} & 0 & 0 & -\Omega_z & 0 \\
 \Psi_z & -\omega_y & \omega_x & 0 & 0 & 0 & 0 & \omega_y & -\omega_x
 \end{bmatrix}
 = \begin{bmatrix}
 \delta\theta_x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \delta\theta_y & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \delta h & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \delta v_x^1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
 \delta v_y^1 & 0 & 0 & (\rho + 2\Omega)_y & 0 & -f_z & f_y & \delta v_x^1 & \delta v_y^1 \\
 \delta v_z^1 & 0 & 0 & (\rho + 2\Omega)_x & f_z & 0 & -f_x & \delta v_y^1 & \Psi_z
 \end{bmatrix}
 + \begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \delta f_x & \delta g_x & 0 & 0 & 0 & 0 & 0 & \epsilon_x & \epsilon_y \\
 \delta f_y & \delta g_y & 0 & 0 & 0 & 0 & 0 & \epsilon_x & \epsilon_z \\
 \delta f_z & \delta g_z & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix}
 \begin{bmatrix}
 0 \\ 0 \\ 0 \\ \epsilon_x \\ \epsilon_y \\ \epsilon_z
 \end{bmatrix}$$

Fig. 13.2 Navigation system error dynamics: computer frame.

This small attitude error model is similar to those presented in Chapter 5.

In addition to navigation state errors, accelerometer and gyro bias errors are included in the Kalman filter state vector. Environment-sensitive process noise is added to velocity and tilt error states to account for unmodeled errors.

### 13.1.4 Observations

Aircraft navigation state positions are used as observations in the transfer-alignment Kalman filter algorithm. The angular position difference is obtained from the following matrix product [see Eq. (13.11)]:

$$(C_n^e)^T (\bar{C}_n^e) = [I - (\Delta \boldsymbol{\theta} \times)]^T \quad (13.20)$$

where, in this case, the  $\bar{C}_n^e$  direction cosine matrix is obtained from the IMU's navigation equation solution subprocess and the  $C_n^e$  matrix is from the aircraft's navigation system. The horizontal angular position observations are used from this product. The altitude observation is the difference between the aircraft's navigation state altitude and the altitude from IMU navigation equations.

The measurement matrix/state vector product, defined for the preceding state vector, becomes

$$Hx = \begin{bmatrix} 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \end{bmatrix} \begin{bmatrix} \delta\theta_x \\ \delta\theta_y \\ \delta h \\ \vdots \end{bmatrix} \quad (13.21)$$

The position observations are processed at a 1-Hz rate.

### 13.1.5 Application of Filter Estimates to Navigation Equations

*Position* updates are expressed as deltas to the position direction cosine matrix and are given by the following product [see Eq. (13.11)]:

$$\Delta C_e^n = -(\delta \hat{\boldsymbol{\theta}} \times) \bar{C}_e^n \quad (13.22)$$

where the caret represents the estimate derived from the alignment filter's outputs. In applying this correction, the assumption  $\delta\theta_z = 0$  is used.

*Attitude* updates based on the  $\phi$  form in Eq. (13.15) are used to correct the direction cosine matrix as

$$\Delta C_b^n = -(\hat{\boldsymbol{\phi}} \times) \bar{C}_b^n \quad (13.23)$$

*Velocity* update estimates are first translated into the true navigation frame. This is accomplished by the following equation:

$$\Delta v^n = \delta \hat{v}^1 - \delta \hat{\boldsymbol{\theta}} \times \bar{v}^n \quad (13.24)$$

Estimates from the Kalman filter algorithm are used to correct the navigation solution at 0.2 Hz.

### 13.1.6 Test Data Overview

Flight test data are used for results presented in this section. These data were recorded during the recently completed U.S. Air Force flight-test program. Data available from this program include an F-16 aircraft's navigation state data and IMU output data, as illustrated in Fig. 13.2.

Results presented in the following sections are from the flight-test program's captive carry flight-test data. The captive tests executed a sequence of maneuvers to line up the aircraft for drop-test vehicle's simulated release. The aircraft is initially near  $30.6^\circ$  latitude and  $-86.7^\circ$  longitude, with an initial heading of approximately  $60^\circ$  from north. There is an initial heading change with a 25-deg aircraft roll angle, a large (approximately 180-deg) heading change with a sustained 40-deg aircraft roll angle, and a final heading change with a  $-20^\circ$  aircraft roll angle. After this simulated release point, the aircraft's trajectory followed a path simulating the drop-test vehicle's free flight.

Shown in Fig. 13.3 are the flight profile's horizontal positions for captive carry test data used for the results presented. Included in this figure are aircraft navigation system position states (F-16) and IMU navigation solution's inertial navigation system (INS), Kalman filter-corrected positions. On this scale, differences between the F-16 and INS positions cannot be seen.

### 13.1.7 Transfer-Alignment Results

Illustrated in the following are the filter's attitude error state estimates during the alignment process, navigation solution improvements after the alignment filter's measurements have been suspended, and body-referenced misalignment estimates.

Shown in Fig. 13.4 are the filter's estimates for the horizontal tilt error  $\phi_x$ . Also shown are the upper and lower  $1-\sigma$  bounds for these estimates, derived from corresponding diagonal elements of the Kalman filter algorithm's error covariance matrix. Superimposed and scaled to fit on this plot are the aircraft's pitch and roll attitudes, indicating the aircraft's maneuvers. The spikes seen in Fig. 13.4 show the alignment filter's measurement update each second for a 5-s time interval, incorporating the resulting error estimates as corrections to

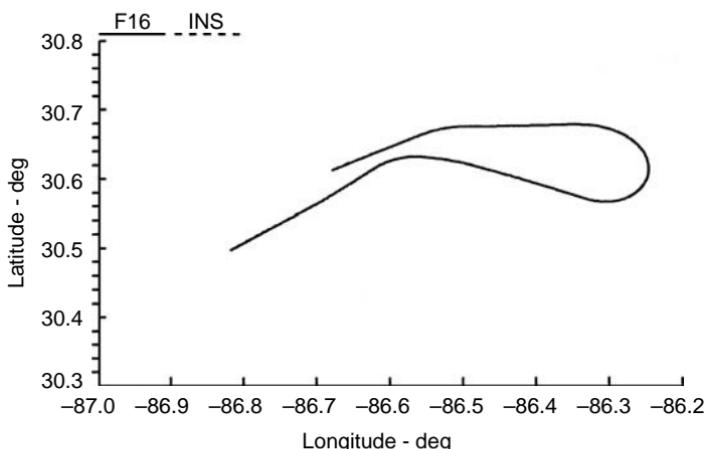
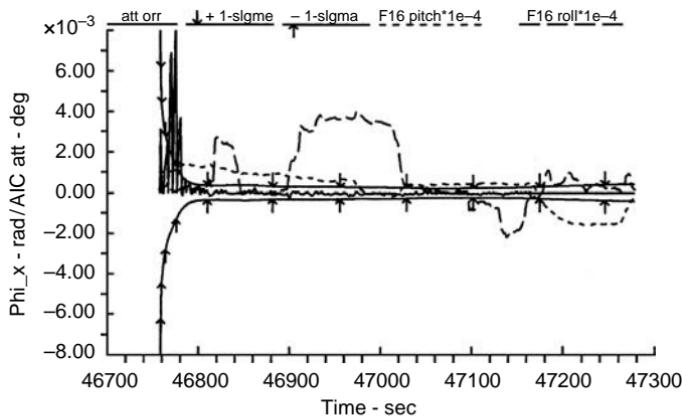


Fig. 13.3 Flight-test profile: horizontal view.

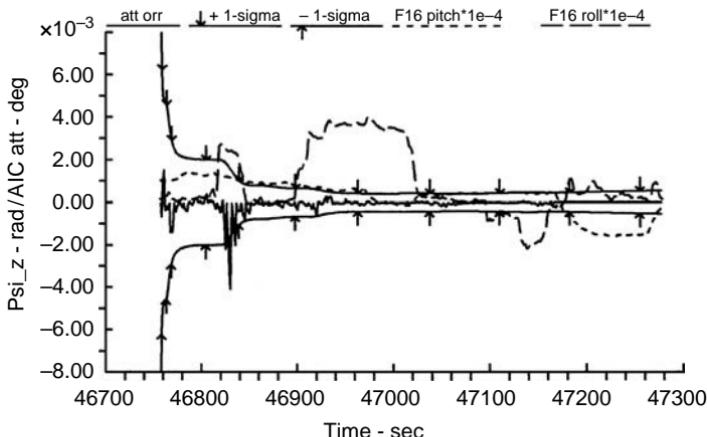


**Fig. 13.4 Filter estimates for  $\phi_x$ .**

the navigation solution, after which the filter states are zeroed for the next 5-s cycle. Figure 13.5 shows estimates for  $\psi_z$ .

The corrected navigation solution's performance is illustrated in Fig. 13.6. In this figure, differences between the aircraft's reference navigation position and the IMU's navigation solution position are shown. The alignment filter's measurements are suspended at approximately 47,165 s during benign aircraft maneuver conditions. Up until that time, the alignment filter's estimates are used to correct navigation equations. After that time, the navigation solution reflects the benefits of those corrections, but without any additional corrections. Examining the postalignment position error growth in this figure, relative to the reference navigation system's position data, provides an additional quality assessment of the alignment filter's estimates.

If the position error growth, after alignment is suspended, is attributed to only horizontal tilt errors, results in Fig. 13.6 indicate that the postalignment attitude alignment error would be less than 1 mrad.



**Fig. 13.5 Filter estimates for  $\psi_z$ .**

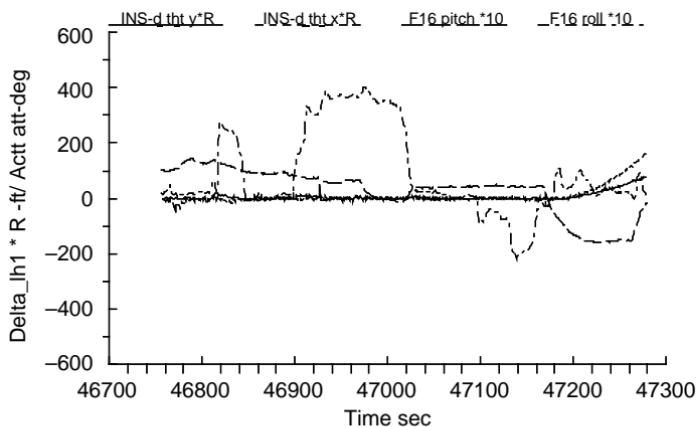


Fig. 13.6 Horizontal position errors.

### 13.1.8 Structural Misalignments

Presented here are results examining body-referenced misalignment errors. These errors are computed from the following matrix product:

$$\begin{aligned} (C_b^n)^T \tilde{C}_b^n &= I - C_n^b (\phi \times) C_b^n \\ &= I - (\phi^b \times) \end{aligned} \quad (13.25)$$

The rationale for comparing INS error model implementation results, using these body-referenced misalignments, is based on near-rigid aircraft assumptions. If the aircraft is rigid, the results from this product should be constant after the filter has converged to its steady-state values. Body-referenced misalignment estimates computed by Eq. (13.25) are shown in Fig. 13.7.

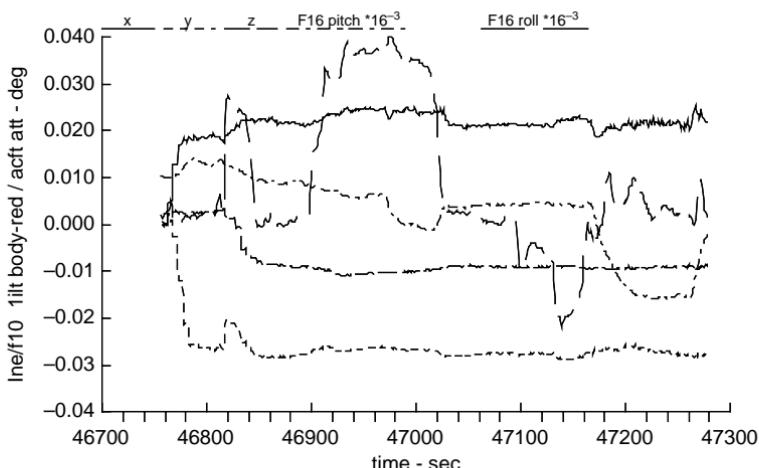


Fig. 13.7 Body-referenced misalignments.

The figure shows the expected near-constant values for body-referenced tilt errors for an assumed rigid structure. However, during maneuvers, a shift is seen from near-constant levels to temporary values (see the  $x$  component), representing a change of several milliradians of attitude.

## 13.2 Alignment Without Benefit of Attitude Initialization [21]

Alignment without initial attitudes is normally the case for stationary ground alignment. A technique to establish initial attitudes, while stationary, using an inertial sensor's accelerometer and gyro outputs was presented in Chapter 11. The problem of in-motion alignment without initial attitudes is less frequently addressed. This problem requires a reformulation of error equations to allow for large attitudes.

In-motion alignment large attitude error model formulations use implementations of two azimuth angle error variables. Examples of these error variables are  $\delta \sin(\alpha)$  and  $\delta \cos(\alpha)$  [22 and 23] and  $\delta \sin(\psi)$  and  $\delta [1-\cos(\psi)]$  [24 and 25]. These formulations assume that products of errors in trigonometric functions and other error variables can again be neglected. In the following, error equations are developed using the first two of the azimuth angle error variables. A perturbation form of INS error equations is obtained. A comparison of this form and that for the second azimuth error variable set is presented in [26].

Data flow for the in-motion alignment problem presented is illustrated in Fig. 13.8. Aircraft navigation system state data are the sources of initial position and velocity data referenced in a geographic frame. The alignment algorithm uses this latitude  $\phi$  and longitude  $\lambda$  in the form of the geographic-to-Earth direction cosine matrix  $C_g^e$  and altitude  $h$  as measurement updates.

### 13.2.1 Navigation Equation Initialization

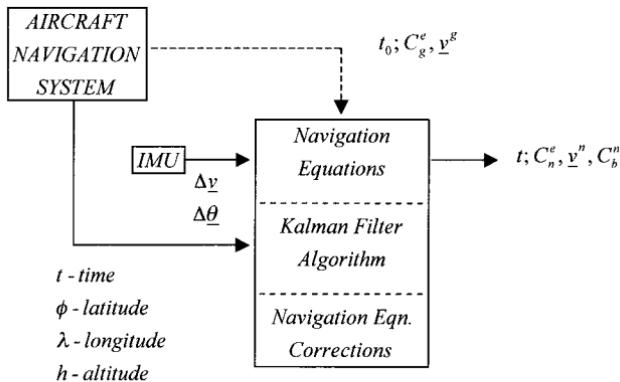
Equations (13.2), (13.5), and (13.9) require initialization before they can be integrated. This is the primary problem for in-motion alignment. Positions (latitude and longitude) are known; therefore, some of the terms/elements in Eq. (13.1) can be computed; however, the wander-azimuth angle is unknown. It is assumed that initial attitudes, roll  $\varphi$ , pitch  $\theta$ , and azimuth  $\psi_{az}$  are unknown. As a result, none of the terms/elements in Eq. (13.7) can be computed.

To address this limitation, the following assumptions are made. Proceeding in the reverse order, it is assumed that initial attitudes are all zero. This assumption produces an identity matrix for the initial value of the  $C_b^n$  matrix. Even though there is a likelihood of large azimuth angles, up to 180 deg, the initial wander-azimuth angle is assumed to be zero. The geographic frame initial velocities are then transformed to the wander-azimuth frame unchanged.

With these assumptions, Eqs. (13.2), (13.5), and (13.9) are numerically integrated. Resulting navigation states will have significant error. It is the task of the alignment algorithm to provide estimates of wander angle and attitude errors that are then used to correct the ongoing integration of navigation equations. Although wander-azimuth errors are large, this process is referred to as coarse alignment.

### 13.2.2 Alignment System Error Equations: System Error Dynamics

*Large-azimuth error model.* Presented in Appendix C is the development of INS system error dynamics for large azimuth errors associated with coarse



**Fig. 13.8 In-motion alignment data flow.**

alignment. This development proceeds with the  $C_e^n$  matrix factored into the following product:

$$C_e^n = C_g^n C_e^g \quad (13.26)$$

The computed  $\bar{C}_e^n$  matrix is expanded into the product of two matrices, each represented in terms of that matrix's errors as

$$\begin{aligned} \bar{C}_e^n &= \bar{C}_g^n \bar{C}_e^g \\ &= (C_g^n + \delta C_g^n)[I - (\delta \theta^g \times)]C_e^g \end{aligned} \quad (13.27)$$

This representation yields the following  $C_e^n$  error matrix:

$$\begin{aligned} \delta C_e^n &= \bar{C}_e^n - C_e^n \\ &= EC_g^n C_e^n \end{aligned} \quad (13.28)$$

Position error is obtained from the  $E$  error matrix

$$E = \begin{bmatrix} \delta c\alpha & \delta s\alpha & -\delta \theta_y \\ -\delta s\alpha & \delta c\alpha & \delta \theta_x \\ \delta \theta_e & -\delta \theta_n & 0 \end{bmatrix} \quad (13.29)$$

Error in the wander-azimuth angle is incorporated as error in the trigonometric sine and cosine functions  $\delta s\alpha$  and  $\delta c\alpha$ , respectively. It is shown in Appendix C that this matrix is governed by the following matrix differential equation:

$$\dot{E} = E\Omega_{e/g}^g - \Omega_{e/n}^n E + (\Omega_{e/n}^n - \bar{\Omega}_{e/n}^n)C_g^n \quad (13.30)$$

The following correspondence exists between the *small-azimuth* error, defined for small-angle error INS system error model forms, and the *large-azimuth* error matrix in Eq. (13.29):

$$-(\delta \theta \times) \sim EC_g^n \quad (13.31)$$

This correspondence is used to obtain velocity and attitude error equations from small-angle INS system error dynamic equations. The INS system error dynamics model for coarse alignment is summarized in Fig. 13.9.

$$\begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \rho_y \\
 0 & -\frac{R^2}{v_x} & 0 & \frac{1}{R} & 0 & 0 & 0 & 0 & -\rho_x \\
 0 & \frac{v_x}{R^2} & -\frac{1}{R} & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
 -(\dot{\psi}_y \Omega_y + 2v_z \Omega_z) & v_y \Omega_x & -\frac{v_z v_x}{R^2} & 2\Omega_z & -(\rho + 2\Omega)_y & -v_y \omega_y & (\dot{\psi}_y \omega_x - f_z) & -2v_z \Omega_x \\
 \dot{\delta\theta}_x^n & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
 \dot{\delta\theta}_y^n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 \frac{d}{dt} \begin{bmatrix} \delta\theta_x^n \\ \delta\theta_y^n \\ \phi_x \\ \phi_y \\ 2\Omega_z \end{bmatrix} = & v_x \Omega_y & -(\dot{\psi}_x \Omega_x + 2v_z \Omega_z) & -\frac{v_y v_z}{R^2} & -2\Omega_z & \frac{v_z}{R} & (\dot{\psi}_x \omega_y + f_z) & -v_x \omega_x & -2v_z \Omega_y \\
 & \dot{\delta\theta}_x^n & 2v_y \Omega_z & \frac{v_x^2 + v_y^2}{R^2} & 2(\rho + \Omega)_x & -2(\rho + \Omega)_y & 0 & -f_y & f_x \\
 & \dot{\delta\theta}_y^n & -\Omega_z & -\frac{v_y}{R^2} & 0 & \frac{1}{R} & 0 & 0 & \Omega_z \\
 & \phi_x & 0 & \frac{v_x}{R^2} & -\frac{1}{R} & 0 & 0 & -\Omega_z & 0 \\
 & \phi_y & 0 & 0 & 0 & 0 & 0 & \omega_y & -\omega_x \\
 & \Omega_z & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & -\omega_y & \omega_x & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix} \times \begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -v_y \epsilon_z & \delta f_x & \delta g_x & 0 & 0 & 0 & 0 & 0 \\
 v_x \epsilon_z & \delta f_y & \delta g_y & 0 & 0 & 0 & 0 & 0 \\
 0 & \delta f_z & \delta g_z & 0 & 0 & 0 & 0 & 0 \\
 \phi_x & 0 & 0 & 0 & 0 & 0 & \epsilon_x & 0 \\
 \phi_y & 0 & 0 & 0 & 0 & 0 & \epsilon_y & 0 \\
 -\delta\alpha & 0 & 0 & 0 & 0 & 0 & \epsilon_z & 0
 \end{bmatrix}$$

Fig. 13.9 INS coarse alignment system error dynamics.

$$\begin{bmatrix}
 0 & 0 & 0 & -\frac{v_y}{R^2} & 0 & \frac{1}{R} & 0 & 0 & 0 & 0 & -\rho_e \\
 0 & 0 & 0 & \frac{v_x}{R^2} & -\frac{1}{R} & 0 & 0 & 0 & 0 & \rho_n & -\rho_e \\
 0 & 0 & 0 & \frac{R^2}{R^2} & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
 -(\dot{\psi}_y \Omega_z + 2v_z \Omega_z) & v_y \Omega_x & -v_z \Omega_x & -\frac{v_z v}{R^2} & 2\Omega_z & -(\rho + 2\Omega)_y & -v_y \omega_y & (\dot{v}_y \omega_x - f_z) & 2v_z \Omega_n & 0 & \delta h \\
 v_x \Omega_y & -(\dot{v}_x \Omega_x + 2v_z \Omega_z) & -\frac{v_y v}{R^2} & -2\Omega_z & \frac{v_z}{R} & (\rho + 2\Omega)_x & (\dot{v}_x \omega_y + f_z) & -v_x \omega_x & 0 & 2v_z \Omega_n & \delta v_x^n \\
 \frac{d}{dt} \frac{\delta \theta_x}{\delta v_x^n} & 2v_y \Omega_z & 2v_y \Omega_z & \frac{v_x^2 + v_y^2}{R^2} & 2(\rho + \Omega)_y & -2(\rho + \Omega)_x & 0 & -f_y & f_x & -2v_x \Omega_n & -2v_y \Omega_n \\
 \phi_x & 0 & -\Omega_z & -\frac{v_y}{R^2} & 0 & \frac{1}{R} & 0 & 0 & \Omega_z & 0 & \Omega_n \\
 \delta s\alpha & \Omega_z & 0 & \frac{v_x}{R^2} & -\frac{1}{R} & 0 & 0 & -\Omega_z & 0 & -\Omega_n & 0 \\
 \delta c\alpha & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix}
 + 
 \begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -v_y \epsilon_z & \delta f_x & \delta g_x & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 v_x \epsilon_z & + \delta f_z & + \delta g_y & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix}$$

Fig. 13.10 INS fine alignment system error dynamics.

During coarse alignment, it is assumed that errors in the trigonometric functions  $\delta s\alpha$  and  $\delta c\alpha$  are constant and that the  $z$ -axis tilt error is zero -  $\phi_z = 0$ .

*Small-azimuth error model.* The perturbation form of INS system error dynamics for fine alignment can be obtained by using the following equations:

$$\delta s\alpha = c\alpha \delta\alpha \quad (13.32)$$

$$\delta c\alpha = -s\alpha \delta\alpha \quad (13.33)$$

It is shown in Appendix D that large-azimuth INS system error dynamics in the preceding subsection collapse to that shown in Fig. 13.10 when the wander-azimuth error angle is small:

This small-azimuth error model form is similar to perturbation error models presented in Chapter 5.

### 13.2.3 Observations

Coarse- and fine-alignment filter models summarized in the preceding subsections, respectively, include angular position error states referenced in the wander-azimuth frame. Measurements are formed to be consistent with these system error dynamic models. In the following, it is assumed that positions are available to form an independent reference source. These positions can be in the form of latitudes, longitudes, and altitudes from either an aircraft navigation system, as in the preceding section, or refined outputs from a GPS receiver.

Consider again the  $C_e^n$  in Eq. (13.1). Define the following vectors from this matrix's columns as

$$\mathbf{t}_3 = \begin{bmatrix} C_{n13}^e \\ C_{n23}^e \\ C_{n33}^e \end{bmatrix} \quad (13.34)$$

and

$$\bar{\mathbf{s}}_1 = \begin{bmatrix} C_{n11}^e \\ C_{n21}^e \\ C_{n31}^e \end{bmatrix} \quad \text{and} \quad \bar{\mathbf{s}}_2 = \begin{bmatrix} C_{n12}^e \\ C_{n22}^e \\ C_{n32}^e \end{bmatrix} \quad (13.35)$$

It was shown in Chapter 5 that the following vector dot products yield differences in the wander-azimuth frame referenced angular position vectors:

$$\Delta\theta_x = \bar{\mathbf{s}}_2 \cdot \mathbf{t}_3 \quad (13.36)$$

$$\Delta\theta_y = -\bar{\mathbf{s}}_1 \cdot \mathbf{t}_3 \quad (13.37)$$

Preceding wander-azimuth frame angular position error differences result in the same measurement matrix/state vector product given in Eq. (13.21):

$$Hx = \begin{bmatrix} 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \end{bmatrix} \begin{bmatrix} \delta\theta_x \\ \delta\theta_y \\ \delta h \\ \vdots \end{bmatrix}$$

### 13.2.4 Alignment Filter Corrections: Position Corrections

The alignment filter algorithm implements system error models summarized in Sec. 13.2.2. Estimates from these algorithms are used to correct navigation equations. These corrections are in the form of position, velocity, attitude, and accelerometer and gyro bias sensor errors.

The position (latitude, longitude, and wander-azimuth) is maintained by the  $C_n^e$  matrix given in Eq. (13.1). The error in the computed  $\bar{C}_n^e$  matrix is represented by Eq. (13.11). Using the following representations for error in the computed latitude, longitude, and wander-azimuth angles (assuming small error angles),

$$\bar{\phi} = \phi + \delta\phi \quad (13.38)$$

$$\bar{\lambda} = \lambda + \delta\lambda \quad (13.39)$$

and

$$\bar{\alpha} = \alpha + \delta\alpha \quad (13.40)$$

and equating the computed  $\bar{C}_n^n$  matrix expressed in terms of its computed latitude, longitude, and preceding wander-azimuth angles to the error form indicated in Eq. (13.11) yield the following expressions for latitude and longitude errors:

$$\delta\phi = -s\alpha\delta\theta_x - c\alpha\delta\theta_y \quad (13.41)$$

and

$$\delta\lambda = \frac{c\alpha\delta\theta_x - s\alpha\delta\theta_y}{c\phi} \quad (13.42)$$

During coarse alignment, the wander-azimuth error  $\delta\alpha$  for navigation state corrections is obtained from the alignment algorithm's states and expressions for  $\delta\alpha$  given in Eqs. (13.32) and (13.33), respectively. This correction is obtained from the following product of the filter's state vector elements as

$$\begin{aligned} c\alpha\delta s\alpha - s\alpha\delta c\alpha &= c\alpha(c\alpha\delta\alpha) - s\alpha(-s\alpha\delta\alpha) \\ &= (s^2\alpha + c^2\alpha)\delta\alpha \end{aligned}$$

Or

$$\delta\alpha = c\alpha\delta s\alpha - s\alpha\delta c\alpha \quad (13.43)$$

During fine alignment, the wander-azimuth error is obtained directly from the alignment algorithm's  $\delta\alpha$  state.

Equations (13.41) and (13.42), and either Eq. (13.43) during coarse alignment or the  $\delta\alpha$  filter state vector element during fine alignment, are used to correct the computed latitude, longitude, and wander-azimuth angles. The corrected  $C_n^e$  matrix given in Eq. (13.1) is recomputed using these corrections.

*Velocity corrections.* Velocity errors are represented, in the perturbation form, by

$$\bar{v}^n = v^n + \delta v^n \quad (13.44)$$

The reverse of this equation is used to apply the corrections from alignment filter(s) estimates.

*Attitude corrections.* Attitude is maintained by the  $C_b^n$  matrix in Eq. (13.7). The error in the computed  $\bar{C}_b^n$  matrix is represented as

$$\bar{C}_b^n = [I - (\phi x)]C_b^n \quad (13.45)$$

During alignment, the z-axis component of the tilt error vector is assumed to be zero. Therefore, the attitude corrections are applied using the reverse of this equation with  $\phi_z = 0$ .

### 13.2.5 In-Motion Alignment Results

The same flight-test data are used for results presented for in-motion alignment and those presented earlier for transfer-alignment. Shown in Fig. 13.11 is the flight profile flown for captive carry test data used for the results presented. Included in this figure are the aircraft navigation state positions (F-16) and the IMU navigation solution's Kalman filter corrected positions during coarse alignment. Slight differences are seen between the F-16 and INS positions during the initial heading change.

The navigation solution's  $C_e^n$  matrix given in Eq. (13.1) is initialized with the aircraft's latitude and longitude, and wander angle is initialized to zero. The attitude initialization for the  $C_b^n$  matrix given in Eq. (13.7) assumes zero for roll  $\varphi$ ,

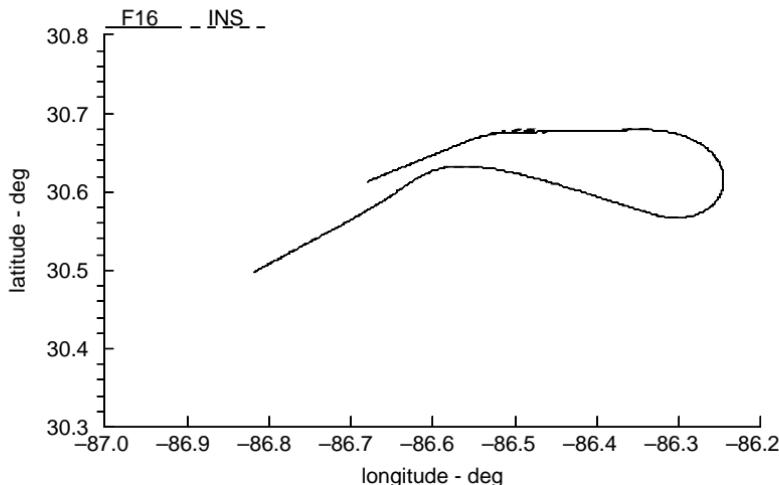
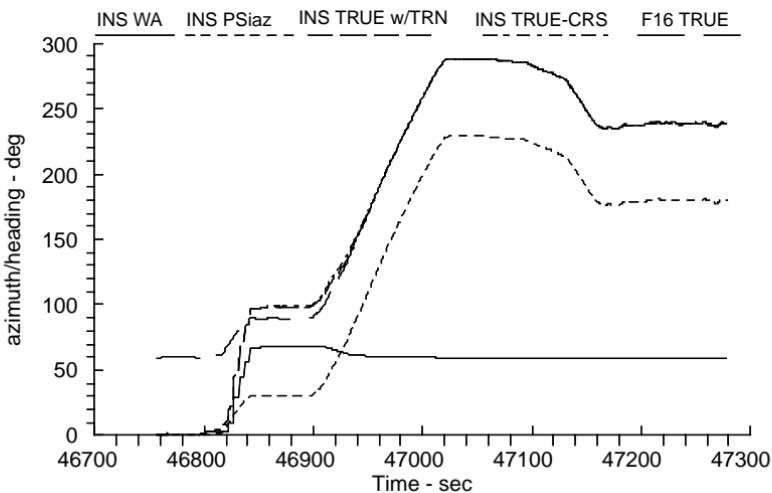


Fig. 13.11 F-16 and INS positions for coarse alignments.



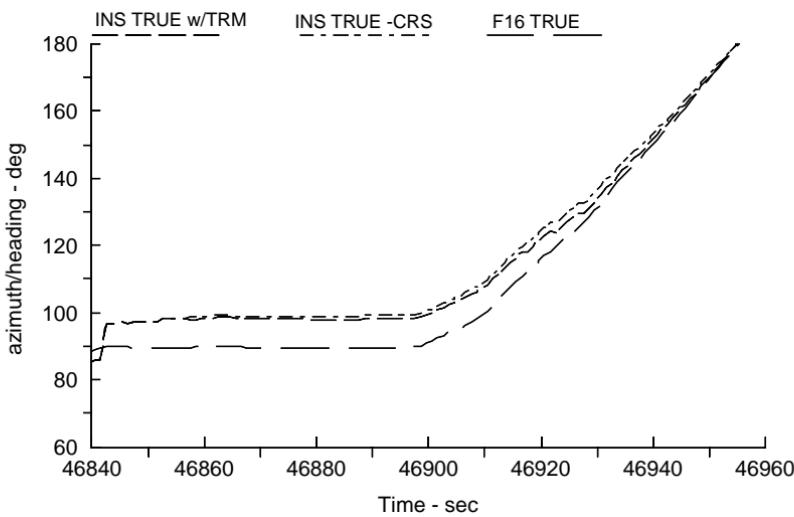
**Fig. 13.12** INS and F-16 true heading for combined and coarse alignments.

pitch  $\theta$ , and azimuth  $\psi_{az}$ . Integrated outputs from navigation equations are processed initially in the coarse-alignment algorithm, then, after some error reduction, in the fine alignment algorithm. Presented in the following are results using the coarse-alignment filter by itself (coarse only) and a combined coarse- and fine-alignment filter (combined). The transition from coarse alignment to fine alignment based on the magnitude of the estimation uncertainty associated with the wander-azimuth error obtained from Eq. (13.43).

The following figures illustrate the benefit of a transition from coarse-alignment to fine-alignment filter models. This transition is based on the coarse-only alignment filter's estimation uncertainty of the wander-azimuth error and occurs for this data set at approximately 46,850 s. Shown in Fig. 13.12 are INS computed azimuths {INS wander angle (INS\_WA), INS azimuth (INS\_PSIaz), and INS true heading (INS\_TRUE w/TRN) using the combined filters; INS true heading (INS\_TRUE-CRS) using the coarse-only filter; and F-16 true heading (F-16\_TRUE)}. From its initial zero value, the INS computed true heading is relatively unchanged until the initial heading change. As a result of this heading change, the INS computed true heading approaches the F-16's true heading—a change of greater than 60 deg. During the second heading change, convergence to the F-16 true heading is nearly complete.

Shown in Fig. 13.13 are the computed INS true heading (INS\_TRUE) results for the initial heading change (expanding the plot scale from that presented in Fig. 13.12). The comparison between combined filters results and those using the coarse-only filter shows a more rapid convergence to the F-16's true heading during the second heading change for the combined filters.

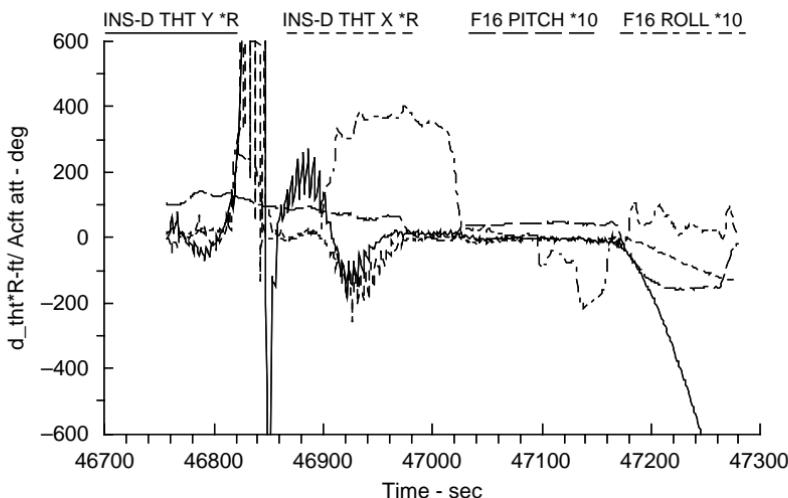
Captive carry tests simulated a drop-test vehicle's free flight near the end of the flight profile. During this simulated free flight, the alignment filter's measurement updates are suspended to allow for an evaluation of unaided navigation



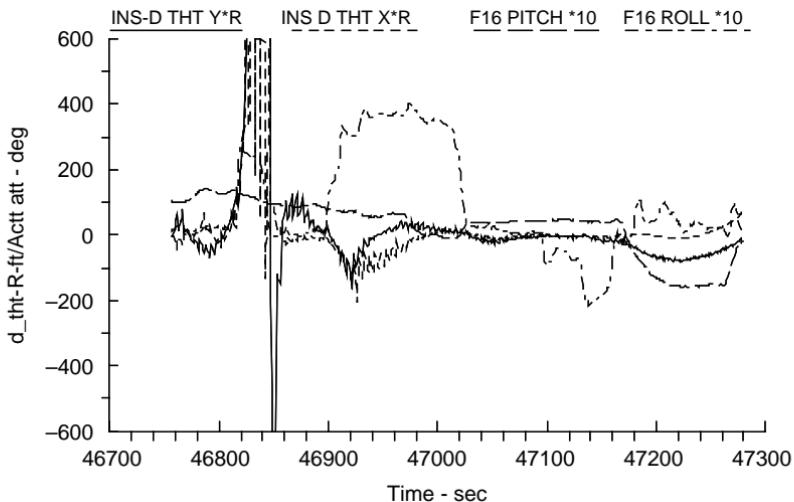
**Fig. 13.13 True headings during initial heading change.**

system errors and, thus, the quality of the alignment achieved. The alignment is suspended at approximately 47,165 s for both filters.

Shown in Figs. 13.14 and 13.15 are position errors  $-\delta\theta_y^*R$  and  $\delta\theta_x^*R$  for combined and coarse-only filters, respectively, for the entire captive test segment. Also presented in these figures are the aircraft's pitch and roll scaled fit onto the same plot. Transient errors during maneuvers are an indication of the degree of the alignment's completeness. Large position error transients



**Fig. 13.14 Position errors for coarse alignment.**



**Fig. 13.15 Position errors for combined alignment with transition.**

occur during the initial heading change while smaller transients are exhibited during later maneuvers, indicating that navigation solutions are being improved using alignment filter(s) corrections. In both figures, after the second heading change, position errors remain small until the alignment process is suspended. After this time, navigation solution is no longer corrected. An examination of these two figures for the period after the alignment is suspended shows a much larger position error growth for coarse-only alignment filter results. These results show the benefit, in terms of a relative performance improvement, of transitioning to fine alignment and using the small attitude error form of the INS system error dynamic model.

This flight-test profile was not designed with the purpose of accomplishing an in-motion alignment but a transfer-alignment. Had in-motion alignment been considered during test planning, a different profile, allowing more settling time between maneuvers, might have yielded better results.

### 13.3 Chapter Summary

In this chapter, two subproblems of IMU in-motion alignment were presented: transfer alignment and alignment without benefit of attitude initialization. The navigation state error model implemented for these two alignment processes differed in that transfer alignment assumes small corrections to reference data and alignment without initialization requires large-azimuth correction capability. The first error model is the same as those presented in Chapter 5, and the second large-azimuth error model was developed in this chapter. For both of these applications, the navigation reference frame is defined in Figs. 3.1 and 3.4.

The transfer-alignment case study shows the benefits of using another navigation system's data for initialization and alignment refinement. By establishing the

alignment with respect to a relatively fixed navigation frame, it is shown that the postalignment attitude alignment error would be less than 1 mrad, even though the relative motion between the IMU and the aircraft's inertial navigation unit (INU) indicates several milliradians of attitude difference during maneuvers caused by aircraft structure flexure. The in-motion alignment without benefit of attitude initialization case study showed the benefits of transitioning to a fine-alignment phase using the small attitude error form of the INS system error dynamic model. These case studies illustrate the importance of the initialization process in generating a navigation solution from differential equations describing navigation states of position, velocity, and attitude. The quality of alignments illustrated can be followed during the Kalman filter's operation. As with the ground alignment case study, filter convergence is enhanced with motion (see Sec. 11.3).

Navigation state equations and error state modeling illustrated for the transfer-alignment case study are functionally similar to other integrated navigation system concepts. Once initialization is complete, "alignment" operations for transfer-alignment are equivalent to "navigation aiding" for a system that has already been aligned (initialized). For example, an INU that has been ground aligned (see Chapter 11) can be assumed to have few remaining errors. For this example, the small attitude error model is applicable. To contain error growth, near continuous updating of the INU's navigation solution is required. An independent source of positions, that is, GPS, can be combined with the INU's navigation state data in a Kalman filter to provide corrections to the INU's navigation solution. This process is functionally similar to the transfer-alignment problem illustrated in this case study.



## Integrated Differential Global Positioning System/Dead-Reckoning Navigation

This chapter presents the sixth application of the elements presented in Part 1. Presented is an integrated dead-reckoning/differential global-positioning-system (DR/DGPS) land navigation system and includes the following segments: dead-reckoning navigation equations, dead-reckoning system error model, DGPS position observations, integrated DR/DGPS implementation, test conditions, and test results.

Dead-reckoning navigation is used for air, land and seacraft. In aircraft, a magnetic compass and an air data system's airspeed sensor can provide the necessary functions to generate a DR navigation solution. On sea, a compass and a speed log provide these functions. Dead-reckoning navigation implements a distance traveled along a travel direction [27]. This navigation implementation assumes simplified horizontal motion and results in a less complex implementation than full three-axis inertial navigation system equations presented in Chapter 5.

The integrated DR/DGPS navigation system presented in this chapter is used to provide accurate continuous navigation state data consisting of position and heading for remotely operated and autonomous land vehicles. Dead-reckoning sensors used in this application are low-cost, fiber-optic rate gyro and radar ground speed sensors. Dead-reckoning navigation provides a stable source of navigation state data for integration with DGPS via Kalman filtering.

While DGPS position data are available, fiber-optic rate gyro and radar ground speed sensors errors are estimated. Several operational scenarios exist in which DGPS data can be disrupted. If DGPS aiding is disrupted, DR-only position and heading navigation data continue until DGPS data are regained.

Results are presented for a real-time implementation of the integrated DGPS/DR navigation system. Kalman filter estimates show the ability to dynamically calibrate low-cost, fiber-optic gyro and radar speed sensor scale-factors. This evaluation is based on comparing the associated position and heading navigation data with qualitative independent references. Evaluations are presented for heading accuracy derived from the numerically integrated rate gyro's corrected outputs and continued positioning accuracy when DGPS is disrupted.

In the following, DR navigation equations are summarized. The Kalman filter implementation for low-cost, fiber-optic rate gyro and speed sensors is presented. Test data demonstrating the position and heading performance with continuous DGPS and with DGPS disrupted are presented.

## 14.1 Dead-Reckoning Navigation Equations

Dead-reckoning navigation dynamic equations are numerically integrated to yield navigation states of position and heading. Illustrated in Fig. 14.1 are the local-level geographic and body-reference frames.

Dead-reckoning position is obtained by numerically integrating geographic velocity components. This incremental distance traveled is referenced to the body frame, which is then transformed into a local-level geographic navigation-frame through the angle  $\Psi$ . This implementation assumes that other vehicle attitudes, that is, roll and pitch, are sufficiently small that they can be ignored.

Consider the following transformation:

$$v^n = C_b^n v^b \quad (14.1)$$

The body-referenced velocity has a nonzero value  $V$  for the first element ( $x$  component), with the other elements zero. This represents the primary direction of travel in body axes. The transformation from the body frame to the geographic frame using transformation matrix  $C_b^n$  is accomplished via the heading angle  $\Psi$ . The transformation indicated in Eq. (14.1) yields geographic-frame velocity components. The latitude  $\phi$  and longitude  $\lambda$  are obtained by numerically integrating their differential equations

$$\dot{\phi} = \frac{v_{\text{north}}}{R_{\text{meridian}} + h} \quad (14.2)$$

and

$$\dot{\lambda} = \frac{v_{\text{east}}}{(R_{\text{normal}} + h) \cos \phi} \quad (14.3)$$

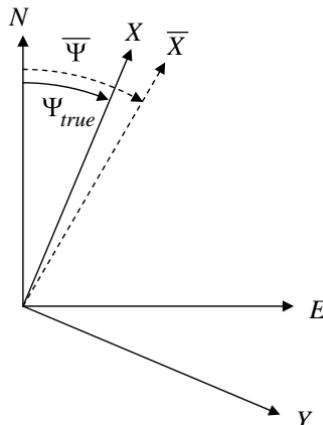


Fig. 14.1 Local-level geographic and body frames.

Heading, used to perform the transformation in Eq. (14.1), is obtained by numerically integrating the output of the rate gyro

$$\Psi = \Psi_{t_0} + \int \dot{\Psi} dt \quad (14.4)$$

This integration of the output of the rate gyro for north-referenced heading ignores the kinematic change of heading as a result of Earth and vehicle transport rates. If operated within a limited region, ignoring contributions from these rates is equivalent to a gyro bias error. An estimate of initial heading  $\Psi_{t_0}$  must be input either from another sensor or based on a prior knowledge.

## 14.2 Dead-Reckoning System Error Model

### 14.2.1 Velocity Error

Figure 14.1 presents errors associated with computed heading. Consider again the transformed navigation-frame velocity from Eq. (14.1):

$$\bar{v}^n = \bar{C}_b^n \bar{v}^b \quad (14.5)$$

where the overbar indicates variables that contain error. The body-referenced velocity is represented as a nominal velocity  $V$  defined such that the  $x$  component is along the primary direction of travel. Velocity errors result from the radar speed sensor's scale-factor  $dSF$ , an  $x$ -axis error, and installation bore-sight  $dBS$ , a  $y$ -axis error. The body-referenced velocity is represented as

$$\bar{v}^b = \begin{bmatrix} V \\ 0 \\ 0 \end{bmatrix} + V \begin{bmatrix} dSF \\ dBs \\ 0 \end{bmatrix} = \begin{bmatrix} (1 + dSF)V \\ dBs V \\ 0 \end{bmatrix} \quad (14.6)$$

Roll  $\varphi$  and pitch  $\theta$  attitude angles used in the  $C_b^n$  transformation matrix in Eq. (14.5) are assumed to be small such that sines and cosines are approximated by the angle in radians and one, respectively. Yaw, or heading  $\Psi$ , is assumed to be composed on the true value plus an error. With these assumptions and using this equation, Eq. (14.5) is rewritten as

$$\begin{aligned} \bar{v}^n &= \bar{C}_b^n \bar{v}^b \\ &\approx \begin{bmatrix} 1 & -\delta\psi & 0 \\ \delta\psi & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\psi & -s\psi & (c\psi\theta + s\psi\varphi) \\ s\psi & c\psi & (s\psi\theta - c\psi\varphi) \\ -\theta & \varphi & 1 \end{bmatrix} \begin{bmatrix} (1 + dSF)V \\ dBs V \\ 0 \end{bmatrix} \\ &= V \begin{bmatrix} 1 & -\delta\psi & 0 \\ \delta\psi & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\psi(1 + dSF) - s\psi dBs \\ s\psi(1 + dSF) + c\psi dBs \\ -\theta(1 + dSF) + \varphi dBs \end{bmatrix} \end{aligned}$$

$$\begin{aligned}
&= V \begin{bmatrix} c\psi(1+dSF) - s\psi dBS - \delta\psi[s\psi(1+dSF) + c\psi dBS] \\ s\psi(1+dSF) + c\psi dBS + \delta\psi[c\psi(1+dSF) - s\psi dBS] \\ -\theta(1+dSF) + \varphi dBS \end{bmatrix} \\
&\approx V \begin{bmatrix} c\psi \\ s\psi \\ 0 \end{bmatrix} + V \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dSF \\ dBS + \delta\psi \\ 0 \end{bmatrix} + V \begin{bmatrix} 0 \\ 0 \\ -\theta \end{bmatrix} \quad (14.7)
\end{aligned}$$

In assuming small roll and pitch angles, the products of these angles and other error variables can be neglected. The navigation-frame velocity error is

$$\delta v^n = V \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dSF \\ dBS + \delta\psi \\ -\theta \end{bmatrix} \quad (14.8)$$

Speed sensor bore-sight and yaw heading errors contribute as a sum to cross-track error. Lumping this sum into an aggregate heading error,

$$\delta\Psi \equiv dBS + \delta\psi \quad (14.9)$$

The navigation-frame velocity error is rewritten as

$$\delta v^n = V \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dSF \\ \delta\Psi \\ -\theta \end{bmatrix} \quad (14.10)$$

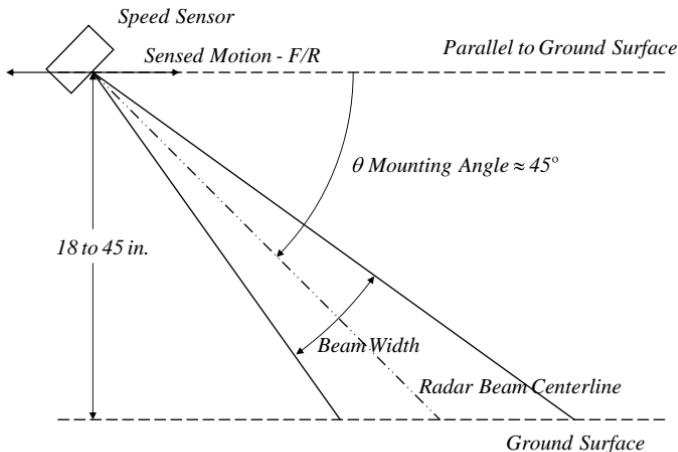
The principal speed sensor error is a scale-factor  $dSF$ . This error's source is the error resulting from the sensor's installation. For the radar ground-speed sensor, the installation is illustrated in Fig. 14.2. The speed sensor's ground-speed output  $v_g$  is related to radar radial velocity  $v_r$  by mounting angle  $\theta$ , resulting from forward/reverse (F/R) sensed motioning.

$$v_g = v_r \cos \theta \quad (14.11)$$

Error in the mounting angle results in the error in measured ground speed

$$\begin{aligned}
\delta v_g &= -v_r \sin \theta \delta\theta \\
&= -v_g \frac{\sin \theta}{\cos \theta} \delta\theta \\
&= -v_g \tan \theta \delta\theta \\
&\approx -v_g \delta\theta \quad (14.12)
\end{aligned}$$

for the manufacturer's recommended 45-deg mounting angle. This demonstrates that the speed sensor's error is proportional to the ground speed, with the proportionality constant being the mounting error angle  $\delta\theta$ .



**Fig. 14.2 Radar ground-speed sensor installation.**

### 14.2.2 Heading Error

The expression for heading error  $\delta\Psi$  in Eq. (14.10) is developed next. Consider again the heading integral in Eq. (14.4). The computed aggregate heading  $\bar{\Psi}$  is represented as

$$\begin{aligned}\bar{\Psi} &= \bar{\Psi}_{t_0} + \int \dot{\bar{\Psi}} dt \\ &= \Psi_{t_0} + \delta\Psi_{t_0} + \int (\dot{\Psi} + \delta\dot{\Psi}_{\text{gyro}}) dt \\ &= \Psi + \delta\Psi_{t_0} + \int \delta\dot{\Psi}_{\text{gyro}} dt\end{aligned}\quad (14.13)$$

The heading error  $\delta\Psi$  becomes

$$\begin{aligned}\delta\Psi &\equiv \bar{\Psi} - \Psi \\ &= \delta\Psi_{t_0} + \int \delta\dot{\Psi}_{\text{gyro}} dt\end{aligned}\quad (14.14)$$

This heading error includes initialization error  $\delta\Psi_{t_0}$  and gyro drift resulting in a heading rate error  $\delta\dot{\Psi}_{\text{gyro}}$ . The gyro drift error is segmented again into a gyro bias drift  $gB$  and scale-factor  $gSF$  contributors. The dynamic equation for the heading error becomes

$$\begin{aligned}\delta\dot{\Psi} &= \delta\dot{\Psi}_{\text{gyro}} \\ &= gB + gSF \dot{\Psi}\end{aligned}\quad (14.15)$$

where  $\dot{\Psi}$  is obtained from the rate gyro's output.

### 14.2.3 Navigation Filter Error Dynamics Summary

Only horizontal error components are included in the DR navigation system error dynamics model. The discrete form of the error propagation equation used to propagate estimates of the filter states from time instant  $i$  to time  $i+1$  is

$$\begin{bmatrix} \delta d_n \\ \delta d_e \\ dSF \\ \delta\psi \\ gB \\ gSF \end{bmatrix}_{i+1} = \begin{bmatrix} 1 & 0 & Vc\Psi\Delta t & -Vs\Psi\Delta t & 0 & 0 \\ 0 & 1 & Vs\Psi\Delta t & Vc\Psi\Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t & \dot{\Psi}\Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta d_n \\ \delta d_e \\ dSF \\ \delta\psi \\ gB \\ gSF \end{bmatrix}_i \quad (14.16)$$

This equation defines the state transition matrix  $\Phi_{i+1,i}$  used for state and covariance matrix time updates.

## 14.3 Differential Global-Positioning-System Position Observations

Differential global-positioning-system positions are used as measurements for the integrated DR/DGPS implementation illustrated. North and east measurements are formed from the differences between DGPS and computed latitudes and longitudes as

$$\Delta d_n = (\phi_{\text{DGPS}} - \bar{\phi})(R_{\text{meridian}} + h_{\text{DGPS}}) \quad (14.17)$$

and

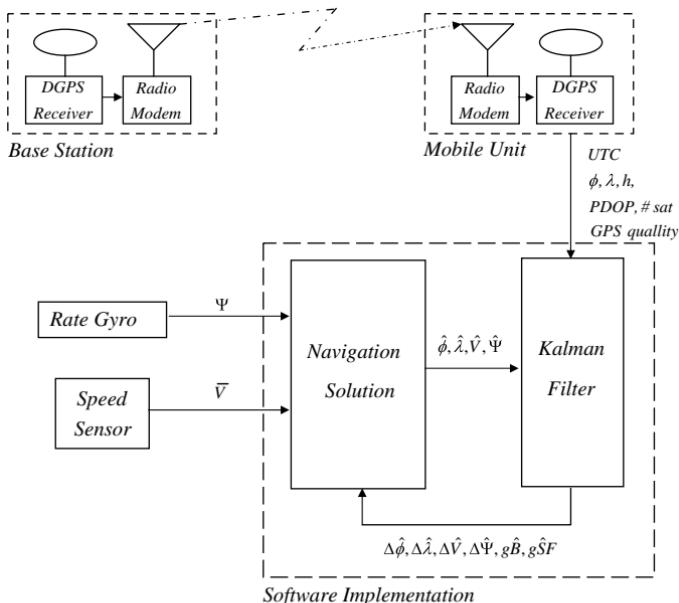
$$\Delta d_e = (\lambda_{\text{DGPS}} - \bar{\lambda})[(R_{\text{normal}} + h_{\text{DGPS}}) \cos \bar{\phi}] \quad (14.18)$$

This form allows the measurement matrix to be expressed as

$$H_{\text{DGPS}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (14.19)$$

## 14.4 Integrated Dead-Reckoning/Differential Global-Positioning-System Implementation

The integrated DR/DGPS implementation functional flow is shown in Fig. 14.3. This figure shows the base and mobile DGPS receivers and the radio modem link between them. Outputs from the mobile DGPS receiver includes UTC time;  $\phi, \lambda, h$  position; position dilution of precision (PDOP); number of satellites; and GPS quality measure. Navigation states are obtained from a numerical integration of the dynamic equations in Eqs. (14.2–14.4). The Kalman filter processes positions as observations comparing navigation state positions and DGPS positions. Estimates of system errors, given in Eq. (14.16), are used to correct ongoing navigation solutions' positions, heading, and rate gyro, and speed sensor inputs.



**Fig. 14.3** Integrated DR/DGPS implementation functional flow.

The processing in Fig. 14.3 is implemented in a laptop personal computer for real-time processing. Navigation states are generated at a data rate of 10 Hz. Differential global-positioning-system position observations are processed in the Kalman filter as measurement updates at a 1-Hz rate.

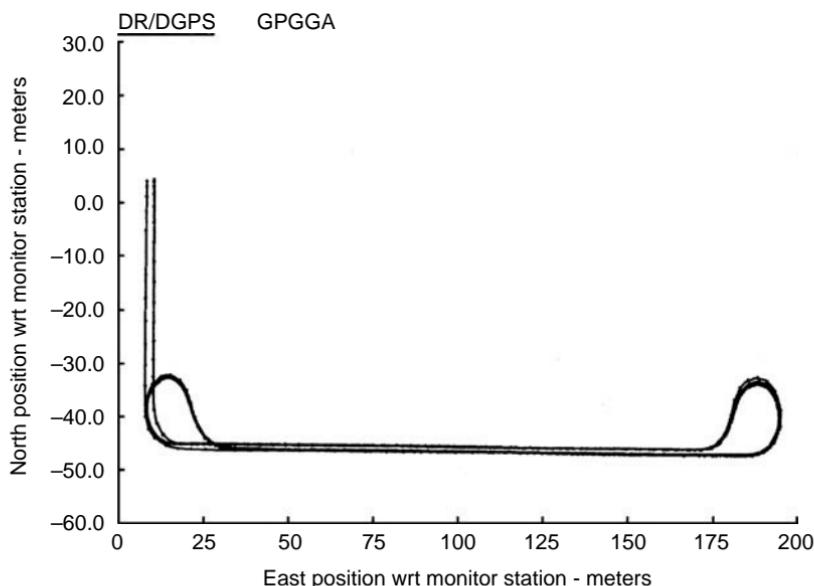
## 14.5 Test Conditions

Results obtained from real-time implementation are presented in Sec. 14.6. The performance is assessed by comparing the navigation state position and heading data with qualitative independent references. A manually operated nonoperational vehicle follows a closed-circuit ground path, retracing this track many times to provide test data for performance evaluations. Ground-track directions approximate cardinal directions: *N–S–E–W*. Positioning accuracy is evaluated based on the repeatability of DR/DGPS positions while the vehicle repeats the ground-track. Heading accuracies are evaluated while the vehicle's track follows cardinal headings. A typical vehicle ground-track followed during testing is presented in Fig. 14.4. Similar tracks were used for tests simulating disrupted DGPS.

## 14.6 Test Results

### 14.6.1 Nominal Differential Global Positioning System

Test results are presented comparing raw differential GPS positions (GPGGA—dots) and DR/DGPS navigation state positions (DR/DGPS—a



**Fig. 14.4 Test ground-track profile.**

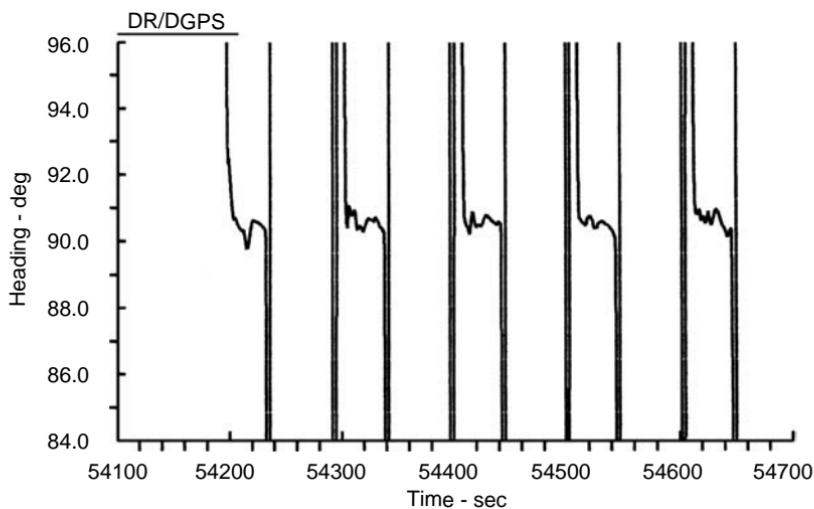
solid line). This test profile's eastward and westward tracks are nearly parallel with true east and west (skewed approximately  $\frac{1}{2}$  deg from true east/west headings) and are separated by approximately 1 m. Dead-reckoning/differential global-positioning-system positions' repeatability is within a few tenths of meters, which is consistent with the DGPS receiver's position accuracies.

Shown in Fig. 14.5 are DR/DGPS navigation state headings (DR/DGPS—a solid line) from the same test presented in Fig. 14.4, with an expanded scale about a 90-deg heading. The headings are seen to vary within a range of approximately 1 deg about the nominal eastward heading value. Going off the figure's vertical scales are headings during the test vehicle's turnarounds. A contribution to this heading variation includes an error caused by manually steering the test vehicle. The computed standard deviation for combined eastward and westward (not shown) heading results is 0.2 deg  $1-\sigma$ .

Shown in Fig. 14.6 are Kalman filter estimates of the speed sensor's scale-factor error  $dSF$  (solid line) and the rate gyro scale-factor error  $gSF$  (dashed line), respectively. The Kalman filter's estimates for these errors have reached a near steady state after a few minutes of operation.

#### 14.6.2 Disrupted Differential Global Positioning System

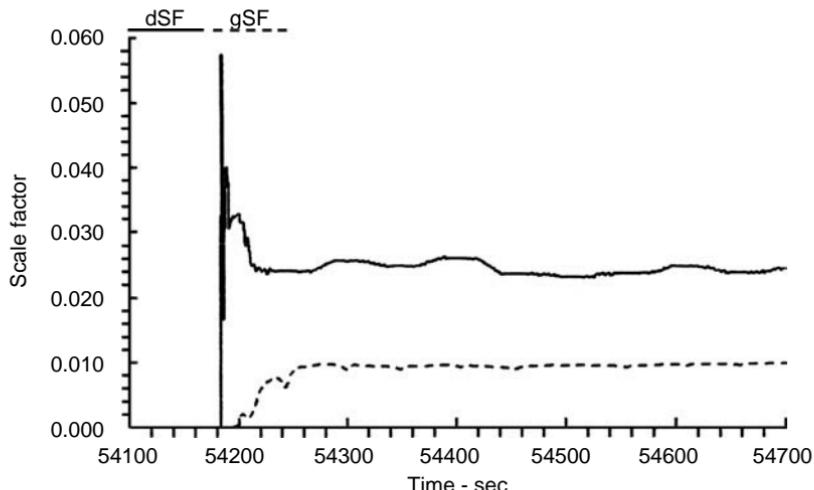
The same test profile used for nominal DGPS operations was used to evaluate the DR/DGPS navigation system's accuracy with disruptions in DGPS data. Positions are shown in Fig. 14.7 (DR/DGPS—solid line and raw DGPS GPGGA positions—dots). The disruption was caused by removing the power to the mobile DGPS receiver's radio modem. The radio modem power was disconnected during the eastward segment at approximately its midway point after



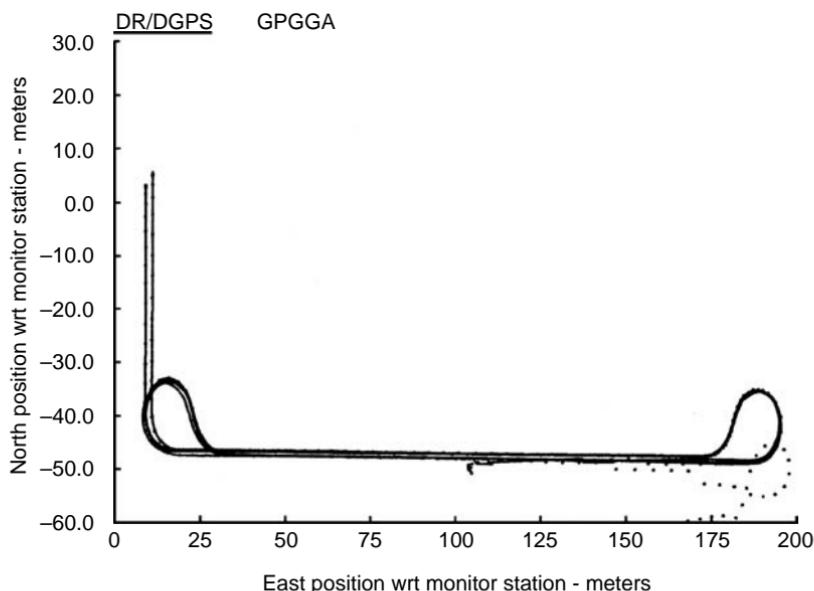
**Fig. 14.5** Eastward headings.

three nominal repeat tracks about this circuit. Eastward and westward segments were completed without DGPS (GPS without differential corrections). The radio modem power was reconnected at approximately the same position where it was disconnected earlier, and the remaining portion of the circuit was completed.

North relative positions with respect to the monitor station are presented in Fig. 14.8 in expanded scale for a time period when modems were disconnected. The change in north (latitude) DGPS position (GPGGA—dots) is seen to occur



**Fig. 14.6** Speed sensor and gyro scale-factor estimates.



**Fig. 14.7** Ground-track profile for disrupted DGPS.

almost immediately after the modem's disconnection at approximately 45,075 s. A few seconds of DGPS data are output past the disconnect time until approximately 45,100 s, and its apparent accuracy is the result of the DGPS receiver's continuing to use the differential corrections available prior to disconnecting the modem. In this figure, the receiver's non-DGPS position data error becomes so large that it falls outside of the figure's axis scale range.

The Kalman filter is implemented to disregard position observations if observed positions fail an internal data-editing test. The receiver's non-DGPS positions (GPGGA—dots) begin to fail this internal test at approximately 45,100 s. When this occurs, DR/DGPS navigation state positions (DR/DGPS—solid line) are DR-only positions. The Kalman filter's corrections improve the speed sensor and gyro data up until non-DGPS data fail the internal data-editing test. The DR-only navigation continues because the DGPS position data are not sufficiently accurate to pass the internal data-editing test.

A single test profile circuit is completed while the radio modem is disconnected. The modem's power is reconnected at approximately 45,205 s. Prior to the radio modem's power being reconnected, the DR-only north (latitude) relative positions agree to within 1 m with the earlier DR/DGPS navigation data just before disconnecting the modem. During the disruption, relative north positions agree with the track's positions before and after disconnecting the modem.

Just after the modem's power is restored, during filter transients while the DGPS receiver is applying corrections received via the modem, DGPS navigation state positions indicate some variation. When DGPS receiver's position data are

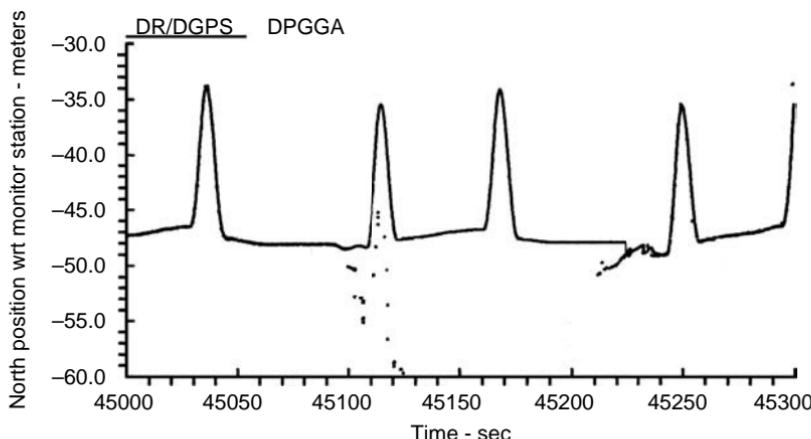


Fig. 14.8 North relative positions with disrupted DGPS.

again within acceptable bounds of the Kalman filter's internal data-editing function, they are accepted as valid positions. After transients subside at approximately 45,225 s, DR/DGPS navigation state positions again accurately follow DGPS positions.

This integrated DR/DGPS implementation, using a low-cost, fiber-optic rate gyro and radar ground-speed sensor, positions—consistent with the DGPS receiver's positioning accuracy and headings and accurate to a few tenths of a degree—can be provided continuously. These continuous data are being provided while offering insensitivity to operational environment variables and disruptions in DGPS.

## 14.7 Chapter Summary

This chapter presented an integrated DR/DGPS land navigation system. This integrated navigation system is used to provide accurate continuous navigation state data consisting of position and heading for remotely operated and autonomous land vehicles. Dead-reckoning sensors used in this application are low-cost, fiber-optic rate gyro and radar ground-speed sensors. Dead-reckoning navigation provides a stable source of navigation state data for integration with DGPS via Kalman filtering.

While DGPS position data are available, fiber-optic rate gyro and radar ground-speed sensors errors are estimated. Several operational scenarios exist in which DGPS data can be disrupted. If DGPS aiding is disrupted, DR-only position and heading navigation data continue until DGPS data are regained.

Results are presented for a real-time implementation of the integrated DGPS/DR navigation system. Kalman filter estimates show that the ability to dynamically calibrate low-cost, fiber-optic gyro and radar speed sensor scale-factors are based on comparing associated position and heading navigation data to qualitative independent references. Evaluations of heading accuracy derived from the

numerically integrated rate gyro's corrected outputs and continued positioning accuracy when DGPS is disrupted are presented.

This DR/DGPS implementation, when operated consistently with the simple horizontal motion assumptions, can provide accurate and fault-tolerant navigation state data. However, as a result of the limited number and functionality of sensors used, navigation state data are limited to horizontal position, velocity, and heading.

## Attitude-Determination and Estimation

In this chapter, the final application of the elements in Part 1, the problems of attitude-determination and estimation are examined. Like the preceding chapter's application, this chapter addresses navigation problems that use a less than complete suite of sensors. In summary the following are presented in this chapter: terrestrial navigation problem of attitude-determination, iteration algorithms for attitude determination, and Kalman filter forms for attitude estimation. In general, the problem of attitude-determination involves determining the transformation matrix (direction cosine matrix) that transforms information from one reference frame to another. Information is sensed in one frame, and a model of this information is assumed in another frame. The DCM that maps the sensed and modeled information is to be determined.

GPS receivers and microelectromechanical-system (MEMS) accelerometers are currently being produced in sufficient quantities that make them affordable. Combining these low-cost sensors, along with magnetic field sensors also low-cost, permits establishing attitude that might be suitable for many terrestrial applications. One such implementation is presented in the following section.

In many applications, it is desired to determine attitude. The preceding discussion is one such application. In these applications, determining attitude involves an iterative process using different means of computing the DCM matrix. Various algorithms, which can be used to compute this matrix, are presented in the second section of this chapter.

Earlier in Chapter 5 and in subsequent discussions, many mathematical forms of error dynamic models were developed and presented for terrestrial inertial navigation systems. These forms were developed for navigation systems that assumed an orthogonal triad of accelerometers and gyros were available to mechanize the navigation system. In space applications, accelerations no longer provide sufficient coupling between the attitude errors and position/velocity errors, and the problem of attitude and position/velocity estimation are effectively decoupled. In the last section of this chapter, different mathematical model forms are developed and presented for attitude estimation via a Kalman filter.

In each of the applications presented in this chapter, numerical results are presented.

### 15.1 Terrestrial Attitude-Determination

GPS receivers generally provide position and velocity data in the form of geographic-referenced latitude, longitude and altitude positions, and velocities.

Attitude is the additional information required. In this example, attitude is determined by using magnetometer and accelerometer sensors and accelerations derived from GPS data. The Earth's magnetic and gravity fields are well known and are characterized in various available forms. This knowledge along with these onboard sensors can be used to determine attitude.

### 15.1.1 DCM Determination

The problem is to determine an unknown direction cosine matrix (DCM) that transforms body-referenced sensor outputs into corresponding geographic frame magnetic and gravity field components. This is expressed by the following equations. For a body-referenced magnetic sensor to match the local geographic-referenced magnetic field, then

$$\mathbf{m}^g = C_b^g \mathbf{m}_{\text{sensed}}^b \quad (15.1)$$

and, for a body-referenced accelerometer to match the local geographic-referenced acceleration,

$$\mathbf{f}^g = C_b^g \mathbf{f}_{\text{sensed}}^b \quad (15.2)$$

Assuming the vectors are not parallel, then a third vector can be formed from the cross product of these vectors. A matrix is formed using the three vectors just described as its columns. The approach described next follows a similar development described in Chapter 11.

The two vector equations, and the third resulting from the vectors' cross product, can be expressed in the following matrix equation:

$$[\mathbf{m}^g \quad \mathbf{f}^g \quad \mathbf{m}^g \times \mathbf{f}^g] = C_b^g [\mathbf{m}_{\text{sensed}}^b \quad \mathbf{f}_{\text{sensed}}^b \quad \mathbf{m}_{\text{sensed}}^b \times \mathbf{f}_{\text{sensed}}^b] \quad (15.3)$$

The matrix on the left-hand side (LHS) is composed of known geographic-referenced information. The bracketed matrix on the right-hand side (RHS) is composed of sensed information. Therefore, the unknown in this equation is the DCM  $C_b^g$ . This equation is manipulated into the following more convenient form:

$$C_b^g = [\mathbf{m}^g \quad \mathbf{f}^g \quad \mathbf{m}^g \times \mathbf{f}^g]^{-T} [\mathbf{m}_{\text{sensed}}^b \quad \mathbf{f}_{\text{sensed}}^b \quad \mathbf{m}_{\text{sensed}}^b \times \mathbf{f}_{\text{sensed}}^b]^T \quad (15.4)$$

Whereas gravity is known, the acceleration must be inferred from information from the vehicle motion and the known gravity. The time rate of change of geographic-referenced velocity is given by the following equation (see Chapter 5):

$$\dot{\mathbf{v}}^g = \mathbf{f}^g - (\boldsymbol{\omega}_{e/g}^g + 2\boldsymbol{\omega}_{l/e}^g) \times \mathbf{v}^g + \mathbf{g}^g \quad (5.18)$$

For slow-moving vehicles, the middle term on the RHS of this equation can be neglected compared to the others. With this assumption, then the geographic-referenced acceleration can be solved for in terms of the velocity derivative and

local geographic-referenced gravity as

$$\mathbf{f}_t^g \approx \dot{\mathbf{v}}^g - \mathbf{g}^g \quad (15.5)$$

The velocity derivative  $\dot{\mathbf{v}}^g$  is not available; however, a numerical derivative can be formed from time sequences of GPS velocity outputs over a time interval  $\Delta t$ , and gravity is computed based on GPS provided latitude and altitude. The following equation is used to approximate the geographic-referenced acceleration at a time instant  $t$ :

$$\mathbf{f}_t^g \approx \frac{(\mathbf{v}_{\text{GPS } t}^g - \mathbf{v}_{\text{GPS } t-\Delta t}^g)}{\Delta t} - \mathbf{g}^g(\text{lat}_{\text{GPS } t}, \text{alt}_{\text{GPS } t}) \quad (15.6)$$

The application of the preceding attitude-determination algorithm is limited to the rate that GPS velocity information is available. This rate might be too low to satisfy the control function requirements.

### 15.1.2 Attitude Change and Observation

Attitude is determined by computing the change in attitude from a previous time instant. After initializing the DCM, a computed DCM is represented by the following:

$$\bar{C}_b^g = [I - (\boldsymbol{\phi} \times)]C_b^g \quad (15.7)$$

where the skew-symmetric matrix  $(\boldsymbol{\phi} \times)$  is composed of the elements of the attitude “change” vector  $\boldsymbol{\phi}$ , where

$$-(\boldsymbol{\phi} \times) \equiv \begin{bmatrix} 0 & \phi_d & -\phi_e \\ -\phi_d & 0 & \phi_n \\ \phi_e & -\phi_n & 0 \end{bmatrix} \quad (15.8)$$

The vector  $\boldsymbol{\phi}$  is used in this application to represent small attitude changes relative to the DCM  $C_b^g$  resulting in the DCM  $\bar{C}_b^g$ .

Consider a vector, with the latter DCM transforming information from the body frame to a geographic frame as

$$\begin{aligned} \bar{\mathbf{u}}^g &= \bar{C}_b^g \mathbf{u}^b \\ &= [I - (\boldsymbol{\phi} \times)]C_b^g \mathbf{u}^b \\ &= \mathbf{u}^g - (\boldsymbol{\phi} \times) \mathbf{u}^g \\ &= \mathbf{u}^g + (\mathbf{u}^g \times) \boldsymbol{\phi} \end{aligned} \quad (15.9)$$

Then the difference vector becomes

$$\Delta \mathbf{u}^g = \bar{\mathbf{u}}^g - \mathbf{u}^g = (\mathbf{u}^g \times) \boldsymbol{\phi} \quad (15.10)$$

This equation is interpreted as an observation equation relating the change in the vector  $\mathbf{u}^g$  to the unknown  $\boldsymbol{\phi}$  vector. In this application, differences in the magnetic field and acceleration are adjoined into a combined observation defined as

$$\mathbf{z} = \begin{bmatrix} \Delta \mathbf{m}^g \\ \Delta \mathbf{f}^g \end{bmatrix} \quad (15.11)$$

and expressed in the following observation equation form

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v} \quad (15.12)$$

where the  $\mathbf{v}$  vector represents additive uncertainty to the observation. The observation matrix defined by the preceding two equations becomes

$$\mathbf{H} = \begin{bmatrix} (\mathbf{m}^g \times) \\ (\mathbf{f}^g \times) \end{bmatrix} \quad (15.13)$$

This form relates the attitude change vector  $\boldsymbol{\phi}$  to the magnetic and accelerometer sensors. This vector is used to form the change in Euler-angles, which is then used to change the DCM (see Problem 3.4 and Table 15.1).

### 15.1.3 Maneuvering Aircraft Simulation Results

In its intended use, the algorithm only has to estimate the change in attitude from one time instant to another. If the data are available at sufficiently high rates, small attitude changes are expected; therefore, few algorithm iterations are required.

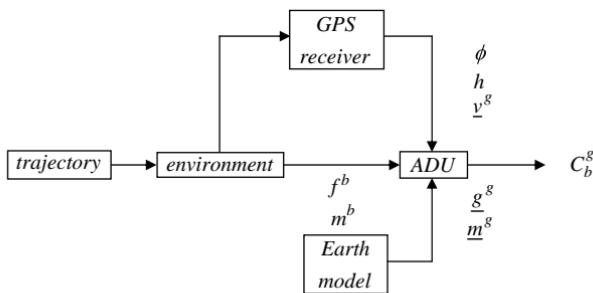
To demonstrate this algorithms performance, numerical results are presented for a maneuvering aircraft application. In Chapters 5 and 12 a simulation model for a maneuvering aircraft was described. That model, shown in Fig. 15.1, is used here to generate simulated accelerometer outputs for use in determining attitude.

This simulation model provides true position, velocity, and attitude for the aircraft's maneuvering environment and generates body-referenced accelerations and magnetic field outputs to represent ideal accelerometer and magnetometer outputs. These ideal sensor outputs are corrupted so that an evaluation can be made as to the impact of sensor error of the algorithms performance. The true values of attitude are then used in this evaluation.

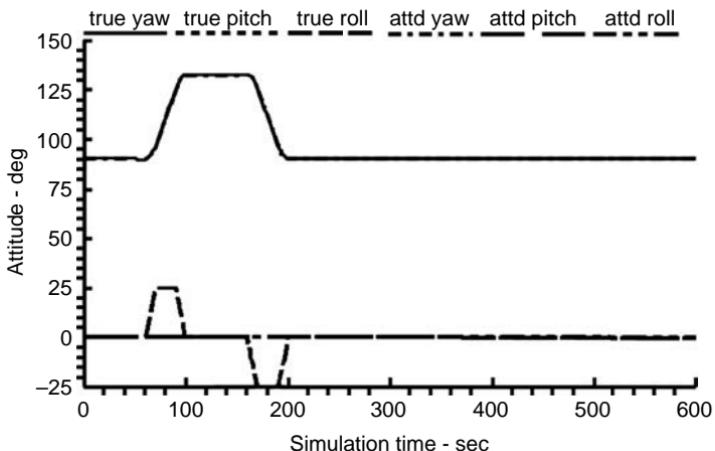
Shown in Fig. 15.2 is a time history of yaw, pitch, and roll values for both the true and determined attitudes. In producing these results, the sensors are not corrupted. The objective is to evaluate the algorithm. The results reproduce the true attitudes with little difference that can be seen on the figure. A higher resolution plot would show stair-step trends in the determined attitudes as a result of the finite rate at which sensor data are available.

Table 15.1 Attitude iteration algorithms

Step	Euler-angle	DCM	Quaternion
1 (Begin)	$H_{\phi} = \begin{bmatrix} (\bar{C}_b^i)^T(s_1^i \times) \\ (\bar{C}_b^i)^T(s_2^i \times) \end{bmatrix}$	$H_{\phi} = \begin{bmatrix} (\bar{C}_b^i)^T(s_1^i \times) \\ (\bar{C}_b^i)^T(s_2^i \times) \end{bmatrix}$	$H_{\phi} = \begin{bmatrix} (\bar{C}_b^i)^T(s_1^i \times) \\ (\bar{C}_b^i)^T(s_2^i \times) \end{bmatrix}$
2	$\hat{\phi} = (H^T H)^{-1} H^T z$	$\hat{\phi} = (H^T H)^{-1} H^T z$	$\hat{\phi} = (H^T H)^{-1} H^T z$
3	$\delta\hat{\psi} = -\hat{\phi}_z - (c\bar{\psi}\hat{\phi}_x + s\bar{\psi}\hat{\phi}_y)t\bar{\theta}$ $\delta\hat{\theta} = s\bar{\psi}\hat{\phi}_x - c\bar{\psi}\hat{\phi}_y$ $\delta\hat{\varphi} = -\frac{(c\bar{\psi}\hat{\phi}_x + s\bar{\psi}\hat{\phi}_y)}{c\bar{\theta}}$	$\delta\hat{C}_b^i = -(\hat{\phi} \times) \bar{C}_b^i$	$\delta q = \frac{1}{2}\bar{Z}^i \hat{\phi}$
4	$\hat{\psi} = \bar{\psi} - \delta\hat{\psi}$ $\hat{\theta} = \bar{\theta} - \delta\hat{\theta}$ $\hat{\varphi} = \bar{\varphi} - \delta\hat{\varphi}$	$\hat{C}_b^i = \bar{C}_b^i - \delta\hat{C}_b^i$	$\hat{q} = \bar{q} - \delta\hat{q}$
5	$\bar{C}_b^i(\hat{\psi}, \hat{\theta}, \hat{\varphi})$	$\bar{C}_b^i = \frac{3}{2}\hat{C}_b^i - \frac{1}{2}\hat{C}_b^i \hat{C}_i^b \hat{C}_b^i$	$\bar{q} = \frac{\hat{q}}{ \hat{q} }$ $\bar{C}_b^i(\bar{q})$
6 (End Return Step 1)	—	—	—



**Fig. 15.1 Attitude-determination simulation flow.**



**Fig. 15.2 True vs determined attitudes.**

#### 15.1.4 Dynamic Model Extensions

A dynamic model can be established for the  $\phi$  vector to include a rate of change of this vector and higher derivatives in a tracking filter. This model can be implemented in a recursive estimator, that is, Kalman filter, to estimate the vector and its rate of change. The rate-of-change estimate is used to extrapolate (predict) a value for this vector between the GPS time instances and create a higher attitude data rate than the GPS base rate. Depending on how rapidly the elements in the  $\phi$  change vector, higher-order dynamical models might be required.

## 15.2 Attitude-Determination by Iteration

In the preceding example, a dynamic nature of the attitude errors was suggested. In this section, static attitude errors are assumed in order to examine several iteration algorithms for determining attitude.

### 15.2.1 Attitude Correction Algorithms

In preceding chapters, several methods for correcting attitude were presented. In Problem 2.13, an iterative algorithm for orthonormalizing a DCM was presented. This algorithm improves a computed DCM that might be slightly corrupted. The concept of an attitude error vector  $\phi$  is introduced. In Problem 3.4, the relationship between the attitude error vector and the Euler-angle errors is presented, and in Problem 3.8 the relationship between this attitude error vector and quaternion errors is presented. These algorithms are summarized in Table 15.1. In the next section, other expressions of attitude error will be presented.

In the following, three methods that use the preceding concepts are compared in estimating a static attitude error. All of these methods first estimate the attitude error vector based on a least-squares solution using two vector observations as introduced in the preceding section. From this estimate, then the three methods either correct the Euler-angles and form a new DCM using the orthonormalizing algorithm or correct the quaternion from which a new DCM is formed. The algorithm using Euler-angles exhibits a singularity at pitch angles whose magnitude is 90 deg and therefore has limitations on its applicability.

### 15.2.2 Iteration Algorithm Comparisons

True attitudes are initially given by the Euler-angles where  $\psi = 90$  deg,  $\theta = 0$  deg, and  $\varphi = 90$  deg, from which a body to inertial transformation matrix  $C_b^i(\psi, \theta, \varphi)$  is computed, and then the corresponding quaternion  $q$  is computed. The iteration algorithms assume that the initial values of the angles are zero:  $\psi = \theta = \varphi = 0$  deg. This implies that the computed DCM is the identity matrix  $C_b(0,0,0) = I$ , and the quaternion is given as  $\bar{q} = [1 \ 0 \ 0 \ 0]^T$ . The two vector observations are formed as the difference between a true vector  $t_{1,2}^b = (C_b^i)^T s_{1,2}^i$  and an estimate of the same vector in a body axis. For the two vectors

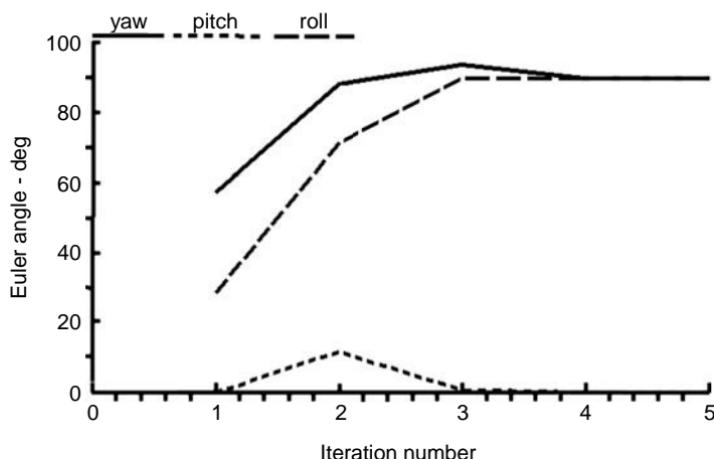


Fig. 15.3 Euler-angle correction iteration algorithm.

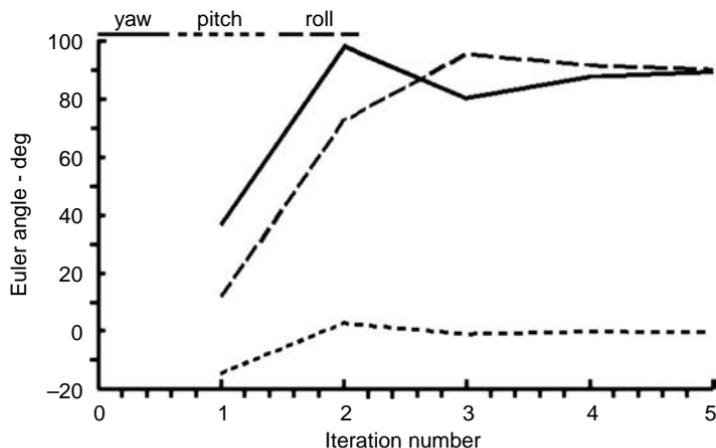


Fig. 15.4 DCM correction iteration algorithm.

these observations are concatenated into the measurement vector

$$\mathbf{z} = \begin{bmatrix} \mathbf{t}_1^b - (\bar{\mathbf{C}}_b^i)^T \mathbf{s}_1^i \\ \mathbf{t}_2^b - (\bar{\mathbf{C}}_b^i)^T \mathbf{s}_2^i \end{bmatrix} \quad (15.14)$$

A non-zero difference between the two vectors is the result of the computed DCM being in error.

The algorithms used are summarized in Table 15.1. The algorithm iterations start at “Begin” and continue through to the “End,” cycling through the algorithm several times in an iterative fashion. Plots algorithms results’ are presented in Figs. 15.3–15.5.

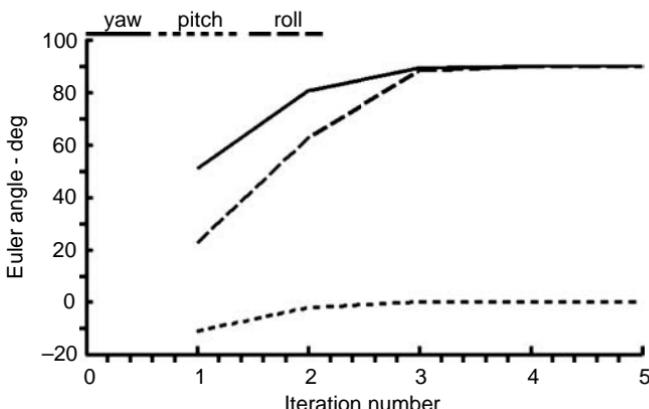


Fig. 15.5 Quaternion correction iteration algorithm.

The results shown in the figures indicate that each algorithm converges to the true Euler-angles from the zero starting values, and the true values are approached within five iteration cycles. The Euler-angle and quaternion correction algorithms show much faster convergence. As a consequence of the Euler-angle correction approach having a mathematical singularity and the relatively slow convergence, the quaternion algorithm is the best of those compared for this static evaluation.

The DCM algorithm, not including the orthonormalization at step 5 and ending at step 4, could have been used to correct the DCM; however, to preserve the orthonormal properties of the DCM, only small corrections would have been permitted, greatly increasing the number of iteration cycles for that form of the algorithm.

### 15.3 Attitude Estimation

In this section, the problem of attitude estimation for spacecraft is presented. The estimation problem can be either to estimate attitude and inertial sensor errors from sensed motion or to estimate attitude and attitude error dynamics from driven motion. In a Kalman filter algorithm implementation, both sensed and driven motion relies on an error dynamic model. Attitude is evolved from the numerical integration of differential equations that describe the attitude dynamics. Being free of the mathematical singularities associated with an Euler-angle dynamics and requiring fewer terms than the DCM matrix differential equation, quaternions are generally used to describe the attitude dynamics.

#### 15.3.1 Attitude Estimation Kalman Filter Error Dynamic Models

Kalman filter algorithm implementations rely on a linear or perturbation form of dynamics equations—error dynamics. These were developed and presented in earlier chapters for terrestrial inertial navigation systems. It has been demonstrated that the error covariance matrix associated with the Kalman filter implementations that use four perturbation quaternion error states is singular (see [28]). As a result, other forms for the quaternion error dynamics have been investigated. In this reference, reduced-state vector representations (three states rather than the four quaternion errors) are related to the quaternion errors. Of those, the body-referenced reduced-state quaternion error vector has been widely used. In [29], this form was applied to inertial rate sensor/optical sensor system alignment calibration. In [30] and [31], a different form, an inertial-referenced reduced-state quaternion error vector, has been applied to attitude dynamics estimation.

In this section, quaternion error vector dynamic models used in current research will be developed to demonstrate their relationship to the original perturbation quaternion error dynamics. In addition to these quaternion error vector forms, the attitude error vector, developed for terrestrial navigation in Chapter 5, will be repeated here in full (see Problem 5.2). Examples of forming the Kalman filter algorithms' observation matrix based on using vector observations, for both the reduced-state quaternion and attitude error vector forms, is presented. The results are summarized in tabular form to ease comparisons between the various error forms.

*Quaternions.* The differential equation describing the quaternion dynamics was given earlier in Chapter 2, Eq. (2.89), as

$$\dot{\mathbf{q}} = -\frac{1}{2} \Omega_{i/b}^b \mathbf{q} \quad (15.15)$$

where angular rotation vector is expanded in the skew-symmetric matrix  $\Omega_{i/b}^b$ . The product of this matrix and the quaternion is expanded and rewritten as

$$\begin{aligned} \Omega_{i/b}^b \mathbf{q} &= \begin{bmatrix} 0 & -\omega_3 & \omega_2 & -\omega_1 \\ \omega_3 & 0 & -\omega_1 & -\omega_2 \\ -\omega_2 & \omega_1 & 0 & -\omega_3 \\ \omega_1 & \omega_2 & \omega_3 & 0 \end{bmatrix}_{\omega=\omega_{i/b}^b} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_0 \end{bmatrix} \\ &= \begin{bmatrix} -q_0 & q_3 & -q_2 \\ -q_3 & -q_0 & q_1 \\ q_2 & -q_1 & -q_0 \\ q_1 & q_2 & q_3 \end{bmatrix}_{\omega=\omega_{i/b}^b} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \equiv Z^b \boldsymbol{\omega}_{i/b}^b \\ &= Z^b C_i^b \boldsymbol{\omega}_{i/b}^i \\ &= \begin{bmatrix} -q_0 & -q_3 & q_2 \\ q_3 & -q_0 & -q_1 \\ -q_2 & q_1 & -q_0 \\ q_1 & q_2 & q_3 \end{bmatrix}_{\omega=\omega_{i/b}^b} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \equiv Z^i \boldsymbol{\omega}_{i/b}^i \quad (15.16) \end{aligned}$$

These expressions define the matrices  $Z^b$  and  $Z^i$  (see Problem 2.11).

*Quaternion error dynamics.* As a result of initialization and other errors, for example, sensor, attitude dynamics uncertainties, etc., a quaternion can become corrupted with error. The corrupted quaternion, represented by an overbar, can be represented as a true value plus an error:

$$\bar{\mathbf{q}} = \mathbf{q} + \delta\mathbf{q} \quad (15.17)$$

The quaternion dynamics equation given by Eq. (15.15) holds for both the corrupted and true values. Substituting this quaternion error representation into the corresponding error forms of Eq. (15.15), subtracting the true value equation from the corrupted equation, and neglecting products of error terms yield the quaternion error dynamics equation.

$$\begin{aligned} \delta\dot{\mathbf{q}} &= -\frac{1}{2} (\delta\Omega_{i/b}^b \mathbf{q} + \Omega_{i/b}^b \delta\mathbf{q}) \\ &= -\frac{1}{2} \Omega_{i/b}^b \delta\mathbf{q} - \frac{1}{2} \delta\Omega_{i/b}^b \mathbf{q} \\ &= -\frac{1}{2} \Omega_{i/b}^b \delta\mathbf{q} - \frac{1}{2} Z^b \delta\boldsymbol{\omega}_{i/b}^b \quad (15.18) \end{aligned}$$

Gyro errors, including bias, scale-factor, misalignment, etc., are contained within the error rate vector  $\delta\omega_{i/b}^b$ . The quaternion error dynamics can also be written, using the preceding definition for  $Z^i$ , as

$$\dot{\delta q} = -\frac{1}{2}\Omega_{i/b}^b \delta q - \frac{1}{2}Z^i \delta\omega_{i/b}^i \quad (15.19)$$

Equations (15.18) and (15.19) describe the quaternion error dynamics that can be used either for sensed or for driven motion.

*Direction cosine matrix.* The equation describing the DCM dynamics is given by the matrix differential equation

$$\dot{C}_b^i = C_b^i(\omega_{i/b}^b \times) = -(\omega_{i/b}^i \times)C_b^i \quad (15.20)$$

where the skew-symmetric matrix equivalent of a vector cross product is defined as

$$(\boldsymbol{\omega} \times) \equiv \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (15.21)$$

*Inertial attitude error vector.* As a result of initialization and other errors, the computed DCM will also be corrupted with error. This error can be represented as the following:

$$\bar{C}_b^g = [I - (\boldsymbol{\phi}^i \times)]C_b^g \quad (15.22)$$

where, in the use in this equation, the inertial attitude error vector  $\boldsymbol{\phi}^i$  represents the misalignment of the computed inertial frame with respect to the true inertial frame. The overbar is used to represent the computed DCM.

*Inertial attitude error vector dynamics.* The error DCM then can be written as the difference between the computed and true DCM as

$$\begin{aligned} \delta C_b^i &= \bar{C}_b^i - C_b^i \\ &= -(\boldsymbol{\phi}^i \times)C_b^i \end{aligned} \quad (15.23)$$

The equation for the attitude error dynamics is obtained by taking the derivative of both rows of this equation and then equating the results. The computed DCM derivative satisfies the same differential equation form as the true DCM

in Eq. (15.20). Taking the derivative of the first row of Eq. (15.23) yields

$$\begin{aligned}
 \delta\dot{C}_b^i &= \bar{C}_b^i - \dot{C}_b^i \\
 &= -(\bar{\omega}_{i/b}^b \times) \bar{C}_b^i + (\boldsymbol{\omega}_{i/b}^b \times) C_b^i \\
 &= -(\bar{\omega}_{i/b}^b \times) [I - (\boldsymbol{\phi}^i \times)] C_b^i + (\boldsymbol{\omega}_{i/b}^b \times) C_b^i \\
 &= -[(\bar{\omega}_{i/b}^b \times) - (\boldsymbol{\omega}_{i/b}^b \times) - (\bar{\omega}_{i/b}^b \times)(\boldsymbol{\phi}^i \times)] C_b^i \\
 &\approx -[(\bar{\omega}_{i/b}^b \times) - (\boldsymbol{\omega}_{i/b}^b \times) - (\boldsymbol{\omega}_{i/b}^b \times)(\boldsymbol{\phi}^i \times)] C_b^i \\
 &\equiv -[(\delta\boldsymbol{\omega}_{i/b}^b \times) - (\boldsymbol{\omega}_{i/b}^b \times)(\boldsymbol{\phi}^i \times)] C_b^i
 \end{aligned} \tag{15.24}$$

Taking the derivative of the second row,

$$\begin{aligned}
 \delta\dot{C}_b^i &= -(\dot{\boldsymbol{\phi}}^i \times) C_b^i - (\boldsymbol{\phi}^i \times) \dot{C}_b^i \\
 &= -(\dot{\boldsymbol{\phi}}^i \times) C_b^i - (\boldsymbol{\phi}^i \times)(\boldsymbol{\omega}_{i/b}^b \times) C_b^i \\
 &= -[(\dot{\boldsymbol{\phi}}^i \times) - (\boldsymbol{\phi}^i \times)(\boldsymbol{\omega}_{i/b}^b \times)] C_b^i
 \end{aligned} \tag{15.25}$$

Equating the preceding results and converting from the skew-symmetric matrix form to vector form yield the following vector differential equation for inertial attitude error vector:

$$\dot{\boldsymbol{\phi}}^i = -C_b^i \delta\boldsymbol{\omega}_{i/b}^b \tag{15.26}$$

This is the same result as in Problem 5.2.

**Body-referenced attitude error vector and dynamics.** A body-referenced attitude error vector can be defined in terms of the preceding inertial attitude error vector as (see Problem 3.7):

$$\boldsymbol{\phi}^b = C_i^b \boldsymbol{\phi}^i \tag{15.27}$$

The error dynamics equation is obtained by taking the derivative of this equation and then substituting equations for the derivatives of each of the terms on the right-hand side from the preceding:

$$\begin{aligned}
 \dot{\boldsymbol{\phi}}^b &= \dot{C}_i^b \boldsymbol{\phi}^i + C_i^b \dot{\boldsymbol{\phi}}^i \\
 &= -(\boldsymbol{\omega}_{i/b}^b \times) C_i^b \boldsymbol{\phi}^i - C_i^b C_b^i \delta\boldsymbol{\omega}_{i/b}^b \\
 &= -(\boldsymbol{\omega}_{i/b}^b \times) \boldsymbol{\phi}^b - \delta\boldsymbol{\omega}_{i/b}^b
 \end{aligned} \tag{15.28}$$

### 15.3.2 Quaternion and Attitude Error Vector Relationships

The body-to-inertial attitude DCM can be expressed in terms of the preceding quaternions as

$$C_b^i = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{15.29}$$

The trace and combinations of off-diagonal elements of the DCM can be used to establish relationships between the quaternions and elements of the DCM (see Problem 2.10). Then, using the DCM error equation, the following relationship between the inertial attitude error vector and the preceding quaternion errors can be established (see Problem 3.8).

$$\begin{bmatrix} \delta q_1 \\ \delta q_2 \\ \delta q_3 \\ \delta q_0 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -q_0 & -q_3 & q_2 \\ q_3 & -q_0 & -q_1 \\ -q_2 & q_1 & -q_0 \\ q_1 & q_2 & q_3 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}_{\boldsymbol{\phi}=\boldsymbol{\phi}^i} \quad (15.30)$$

This equation is rewritten in more compact form using the definition of the preceding matrix  $Z^i$ :

$$\delta \mathbf{q} = \frac{1}{2} Z^i \boldsymbol{\phi}^i \quad (15.31)$$

and, in terms of the body-referenced attitude error vector,

$$\begin{aligned} \delta \mathbf{q} &= \frac{1}{2} Z^i \boldsymbol{\phi}^i \\ &= \frac{1}{2} Z^i C_b^i \boldsymbol{\phi}^b \\ &= \frac{1}{2} Z^b \boldsymbol{\phi}^b \end{aligned} \quad (15.32)$$

These two equations illustrate the relationship between the quaternion error and the two attitude error vector forms.

### 15.3.3 Alternate Inertial Attitude Error Vector Dynamics Equation Development

The inertial attitude error vector equations just developed started from the DCM error equation. To demonstrate the relationship between this error and the perturbation quaternion forms, this equation will be developed from the quaternion and perturbation quaternion equations.

The  $Z$  matrices satisfy the following identities (see Problem 2.11):

$$Z^{b^T} Z^b = Z^{i^T} Z^i = I_{3 \times 3} \quad (15.33)$$

Using this relationship, the inertial attitude error vector is written as

$$\boldsymbol{\phi}^i = 2 Z^{i^T} \delta \mathbf{q} \quad (15.34)$$

The derivative of this equation is used to form the inertial attitude error vector dynamics. This derivative requires the derivative of the matrix  $Z^i$ . Differentiation

of the equation for  $Z^i$  and then substituting the corresponding relationships for the quaternion derivatives yield the following:

$$\dot{Z}^i = \frac{1}{2} \begin{Bmatrix} q_1\omega_1 + q_2\omega_2 + q_3\omega_3 & -q_2\omega_1 + q_1\omega_2 - q_0\omega_3 & -q_3\omega_1 + q_0\omega_2 + q_1\omega_3 \\ q_2\omega_1 - q_1\omega_2 + q_0\omega_3 & q_1\omega_1 + q_2\omega_2 + q_3\omega_3 & -q_0\omega_1 - q_3\omega_2 + q_2\omega_3 \\ q_3\omega_1 - q_0\omega_2 - q_1\omega_3 & q_0\omega_1 + q_3\omega_2 - q_2\omega_3 & q_1\omega_1 + q_2\omega_2 + q_3\omega_3 \\ q_0\omega_1 + q_3\omega_2 - q_2\omega_3 & -q_3\omega_1 + q_0\omega_2 + q_1\omega_3 & q_2\omega_1 - q_1\omega_2 + q_0\omega_3 \end{Bmatrix}_{\omega=\omega_{i/b}^i} \quad (15.35)$$

This can be rewritten in more compact form as

$$\dot{Z}^i = \frac{1}{2} \left[ \mathbf{q} \boldsymbol{\omega}_{i/b}^{i^T} - Z^i (\boldsymbol{\omega}_{i/b}^i \times) \right] \quad (15.36)$$

Returning to the inertial attitude error vector equation, and taking the derivative, substituting the preceding equations for quaternion error dynamics and differential equation for  $Z^i$  yields the following:

$$\begin{aligned} \dot{\boldsymbol{\phi}}^i &= 2 \{ \dot{Z}^{i^T} \delta \mathbf{q} + Z^{i^T} \delta \dot{\mathbf{q}} \} \\ &= 2 \left\{ \frac{1}{2} [\mathbf{q} \boldsymbol{\omega}_{i/b}^{i^T} - Z^i (\boldsymbol{\omega}_{i/b}^i \times)]^T \delta \mathbf{q} + Z^{i^T} \left( -\frac{1}{2} \right) [\Omega_{i/b}^b \delta \mathbf{q} + Z^i \delta \boldsymbol{\omega}_{i/b}^b] \right\} \\ &= \{ \boldsymbol{\omega}_{i/b}^i \mathbf{q}^T + (\boldsymbol{\omega}_{i/b}^i \times) Z^{i^T} - Z^{i^T} \Omega_{i/b}^b \} \delta \mathbf{q} - Z^{i^T} Z^i \delta \boldsymbol{\omega}_{i/b}^i \end{aligned} \quad (15.37)$$

and using the following relationship

$$\{ \boldsymbol{\omega}_{i/b}^i \mathbf{q}^T + (\boldsymbol{\omega}_{i/b}^i \times) Z^{i^T} - Z^{i^T} \Omega_{i/b}^b \} = [0]_{3 \times 4} \quad (15.38)$$

the inertial attitude error vector dynamics equation is obtained:

$$\dot{\boldsymbol{\phi}}^i = -\delta \boldsymbol{\omega}_{i/b}^i = -C_b^i \delta \boldsymbol{\omega}_{i/b}^b \quad (15.39)$$

This is the same as obtained in Eq. (15.26) starting with the DCM error equation.

### 15.3.4 Vector Observation Using Inertial Attitude Error Vector State

Vector observations are formed as the difference between body-mounted sensor measured outputs and the same observations formed from onboard

computed variables. The observation vector becomes

$$\begin{aligned}
 z &\equiv \bar{s}_m^b - \bar{s}^b \\
 &= \bar{M}(s^b + v) - \bar{C}_i^b s^i \\
 &= [I + (\boldsymbol{\delta} \times)](s^b + v) - C_i^b [I + (\boldsymbol{\phi}^i \times)] s^i \\
 &\approx s^b - C_i^b s^i + (\boldsymbol{\delta} \times) s^b - C_i^b (\boldsymbol{\phi}^i \times) s^i + v \\
 &= -(s^b \times) \boldsymbol{\delta} + C_i^b (s^i \times) \boldsymbol{\phi}^i + v
 \end{aligned} \tag{15.40}$$

where the sensor misalignment matrix  $\bar{M}$  and corrupted inertial-to-body DCM  $\bar{C}_i^b$  are represented as

$$\bar{M} \equiv [I + (\boldsymbol{\delta} \times)] \tag{15.41}$$

$$\bar{C}_i^b = C_i^b [I + (\boldsymbol{\phi}^i \times)] \tag{15.42}$$

and products of errors are neglected. The corresponding measurement matrix for this observation is written as this matrix and state vector product as

$$H_{\boldsymbol{\phi}} \mathbf{x} = \begin{bmatrix} C_i^b (s^i \times) & \cdots & -(C_i^b s^i) \times \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}^i \\ \vdots \\ \boldsymbol{\delta} \end{bmatrix} \tag{15.43}$$

### 15.3.5 Perturbation Quaternion Vectors

*Body-referenced quaternion error vector.* The reduced-state body-referenced quaternion error vector is defined by the following (see [28]):

$$\delta q^b \equiv Z^{b^T} \delta q \tag{15.44}$$

To obtain its dynamics equation, the dynamics for  $Z^b$  is required. Again, differentiating the equation for  $Z^b$ , and then substituting the corresponding relationships for the quaternion derivatives yield the following:

$$\dot{Z}^b = \frac{1}{2} \left\{ \begin{array}{ccc} q_1 \omega_1 + q_2 \omega_2 + q_3 \omega_3 & -q_2 \omega_1 + q_1 \omega_2 + q_0 \omega_3 & -q_3 \omega_1 - q_0 \omega_2 + q_1 \omega_3 \\ q_2 \omega_1 - q_1 \omega_2 - q_0 \omega_3 & q_1 \omega_1 + q_2 \omega_2 + q_3 \omega_3 & q_0 \omega_1 - q_3 \omega_2 + q_2 \omega_3 \\ q_3 \omega_1 + q_0 \omega_2 - q_1 \omega_3 & -q_0 \omega_1 + q_3 \omega_2 - q_2 \omega_3 & q_1 \omega_1 + q_2 \omega_2 + q_3 \omega_3 \\ q_0 \omega_1 - q_3 \omega_2 + q_2 \omega_3 & q_3 \omega_1 + q_0 \omega_2 - q_1 \omega_3 & -q_2 \omega_1 + q_1 \omega_2 + q_0 \omega_3 \end{array} \right\}_{\boldsymbol{\omega}=\boldsymbol{\omega}_{i/b}^b} \tag{15.45}$$

This can be rewritten in more compact form as

$$\dot{Z}^b = \frac{1}{2} [Z^b (\boldsymbol{\omega}_{i/b}^b \times) + \mathbf{q} \boldsymbol{\omega}_{i/b}^{b^T}] \quad (15.46)$$

*Body-referenced quaternion error vector dynamics.* Returning to the reduced-state body-referenced quaternion error vector equation, differentiating and substituting the preceding equations for quaternion error dynamics and the differential equation for  $Z^b$  yield the following:

$$\begin{aligned} \dot{\delta\mathbf{q}}^b &= \dot{Z}^{b^T} \delta\mathbf{q} + Z^{b^T} \dot{\delta\mathbf{q}} \\ &= \frac{1}{2} [-(\boldsymbol{\omega}_{i/b}^b \times) Z^{b^T} + \boldsymbol{\omega}_{i/b}^b \mathbf{q}^T] \delta\mathbf{q} + Z^{b^T} \left( -\frac{1}{2} \right) [\Omega_{i/b}^b \delta\mathbf{q} + Z^b \delta\boldsymbol{\omega}_{i/b}^b] \\ &= -\frac{1}{2} [(\boldsymbol{\omega}_{i/b}^b \times) Z^{b^T} + Z^{b^T} \Omega_{i/b}^b - \boldsymbol{\omega}_{i/b}^b \mathbf{q}^T] \delta\mathbf{q} - \frac{1}{2} Z^{b^T} Z^b \delta\boldsymbol{\omega}_{i/b}^b \\ &= -\frac{1}{2} [(\boldsymbol{\omega}_{i/b}^b \times) Z^{b^T} + (\boldsymbol{\omega}_{i/b}^b \times) Z^{b^T}] \delta\mathbf{q} - \frac{1}{2} \delta\boldsymbol{\omega}_{i/b}^b \\ &= -(\boldsymbol{\omega}_{i/b}^b \times) Z^{b^T} \delta\mathbf{q} - \frac{1}{2} \delta\boldsymbol{\omega}_{i/b}^b \\ &= -(\boldsymbol{\omega}_{i/b}^b \times) \delta\mathbf{q}^b - \frac{1}{2} \delta\boldsymbol{\omega}_{i/b}^b \end{aligned} \quad (15.47)$$

*Inertial-referenced quaternion error vector dynamics.* The reduced-state inertial-referenced quaternion error vector is defined in a similar manner as was the body-referenced quaternion error vector by the following:

$$\delta\mathbf{q}^i \equiv Z^{i^T} \delta\mathbf{q} \quad (15.48)$$

Without the factor of 2, this is identical to the inertial attitude error vector definition. Accounting for this factor of 2, the dynamics of this equation can be written directly as

$$\dot{\delta\mathbf{q}}^i = -\frac{1}{2} \delta\boldsymbol{\omega}_{i/b}^i = -\frac{1}{2} C_b^i \delta\boldsymbol{\omega}_{i/b}^b \quad (15.49)$$

*Vector observations using body-referenced quaternion error vector.* The body-referenced quaternion error vector is related to the inertial attitude error vector as

$$\begin{aligned} \delta\mathbf{q}^b &= Z^{b^T} \delta\mathbf{q} \\ &= \frac{1}{2} Z^{b^T} Z^i \boldsymbol{\phi}^i \end{aligned} \quad (15.50)$$

Using the following relationship

$$Z^{b^T} Z^i = C_i^b \quad (15.51)$$

then

$$\delta\mathbf{q}^b = \frac{1}{2} C_i^b \boldsymbol{\phi}^i \quad (15.52)$$

or

$$\boldsymbol{\phi}^i = 2C_b^i \delta \mathbf{q}^b \quad (15.53)$$

Forming the vector observations as just shown from onboard computed variables and using this relationship between the attitude error vector and body-referenced quaternion error vector, the observation vector is expressed as

$$\begin{aligned} \mathbf{z} &\equiv \tilde{\mathbf{s}}_m^b - \tilde{\mathbf{s}}^b \\ &= -(\mathbf{s}^b \times) \boldsymbol{\delta} + C_i^b (\mathbf{s}^i \times) \boldsymbol{\phi}^i + \mathbf{v} \\ &= 2C_i^b (\mathbf{s}^i \times) C_b^i \delta \mathbf{q}^b - (\mathbf{s}^b \times) \boldsymbol{\delta} + \mathbf{v} \\ &= 2(\mathbf{s}^b \times) \delta \mathbf{q}^b - (\mathbf{s}^b \times) \boldsymbol{\delta} + \mathbf{v} \end{aligned} \quad (15.54)$$

In the last step, the following similarity transformation is used:

$$(\mathbf{s}^b \times) = C_i^b (\mathbf{s}^i \times) C_b^i \quad (15.55)$$

The corresponding measurement matrix for this observation is written as this matrix and state vector product as

$$H_{\delta \mathbf{q}^b} \mathbf{x} = \begin{bmatrix} 2(\mathbf{s}^b \times) & \cdots & -(\mathbf{s}^b \times) \end{bmatrix} \begin{bmatrix} \delta \mathbf{q}^b \\ \vdots \\ \boldsymbol{\delta} \end{bmatrix} \quad (15.56)$$

### 15.3.6 Quaternion Corrections from Quaternion Error Vector Forms

The quaternion estimates are corrected based on a filter's estimates of the quaternion error vector. The relationship between the reduced-state body-referenced quaternion error vector and the quaternion error is given by

$$\begin{aligned} \delta \mathbf{q} &= \frac{1}{2} Z^i 2C_b^i \delta \mathbf{q}^b \\ &= Z^i C_b^i \delta \mathbf{q}^b \\ &= Z^b \delta \mathbf{q}^b \end{aligned} \quad (15.57)$$

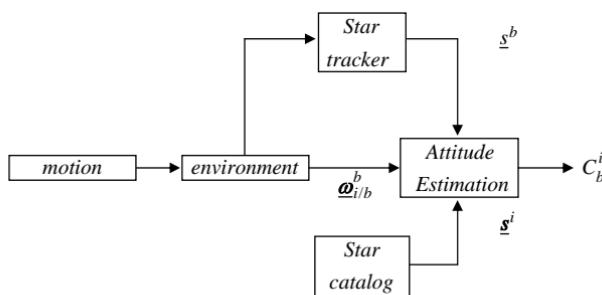
The corresponding relationship for the reduced-state inertial-referenced quaternion error vector and quaternion error is

$$\delta \mathbf{q} = Z^i \delta \mathbf{q}^i \quad (15.58)$$

The error dynamic equation forms just developed are summarized in Table 15.2. Estimates from a Kalman filter algorithm's implementation, using these models, would be used to correct the ongoing numerical integration of

Table 15.2 Error model summary

Error model form	Error dynamics equation	Vector observation matrix	Relationship to quaternion correction
Inertial attitude error	$\dot{\phi}^i = -C_b^i \delta\omega_{i/b}^b$	$H_{\delta\phi} \mathbf{x} = [C_i^b (\mathbf{s}^i \times) \quad \dots \quad -(C_i^b \mathbf{s}^i) \times]$	$\delta\mathbf{q} = \frac{1}{2} Z^i \boldsymbol{\phi}^i$
Body attitude error	$\dot{\boldsymbol{\phi}}^b = -(\boldsymbol{\omega}_{i/b}^b \times) \boldsymbol{\phi}^b - \delta\boldsymbol{\omega}_{i/b}^b$	—	$\delta\mathbf{q} = \frac{1}{2} Z^b \boldsymbol{\phi}^b$
Quaternion error	$\dot{\delta\mathbf{q}} = -\frac{1}{2} \Omega_{i/b}^b \delta\mathbf{q} - \frac{1}{2} Z^b \delta\omega_{i/b}^b$	—	—
Reduced-state body quaternion error	$\dot{\delta\mathbf{q}}^b = -(\boldsymbol{\omega}_{i/b}^b \times) \delta\mathbf{q}^b - \frac{1}{2} \delta\omega_{i/b}^b$	$H_{\delta\mathbf{q}^b} \mathbf{x} = [2(\mathbf{s}^b \times) \quad \dots \quad -(\mathbf{s}^b \mathbf{x})]$	$\delta\mathbf{q}^b = Z^b \delta\mathbf{q}^b$
Reduced-state inertial quaternion error	$\dot{\delta\mathbf{q}}^i = -\frac{1}{2} C_b^i \delta\omega_{i/b}^b$	—	$\delta\mathbf{q} = Z^i \delta\mathbf{q}^i$

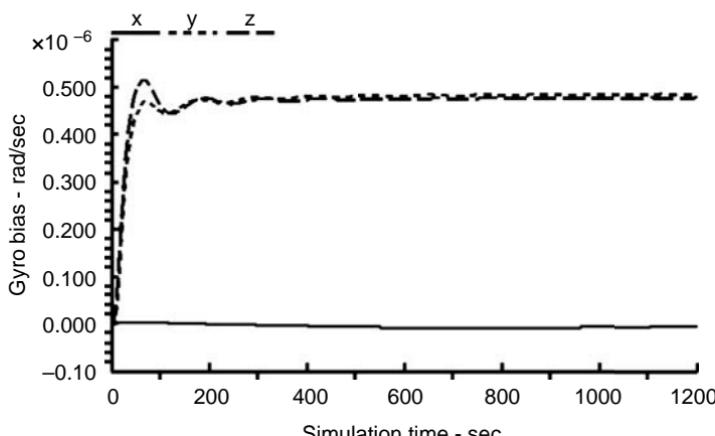


**Fig. 15.6 Attitude estimation simulation flow.**

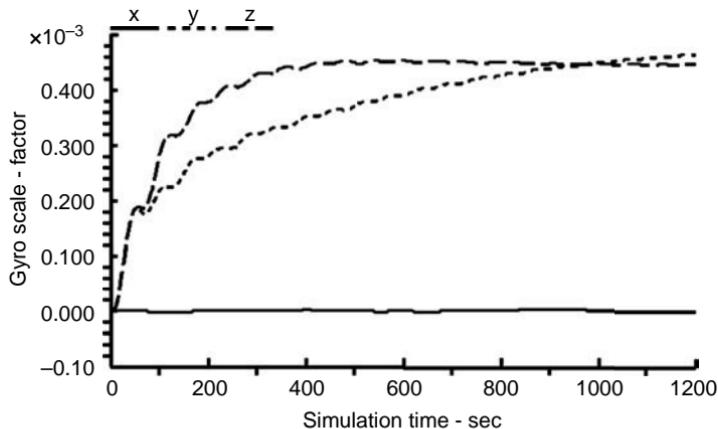
the quaternion dynamics equation given in Eq. (15.15). To accomplish this correction, relationships between that error and the desired quaternion correction are provided.

### 15.3.7 Attitude Estimation Simulation Comparisons

The filter error model forms listed in Table 15.2 are linearly related to each other. In obtaining these forms, assumptions like those used to obtain error models in Chapter 5 for the terrestrial navigation systems that use the wander-azimuth navigation frame were not used. The only assumption in obtaining these models is that products of errors can be neglected. Therefore, it is expected that results obtained by using any of these models should be the same. This similarity is examined using simulations. Presented in this section are results from using inertial attitude error  $\phi^i$  model and the body-referenced reduced-state quaternion error  $\delta q^b$  model evaluated using the same simulated spacecraft motion and observations. Figure 15.6 presents this simulation flow.



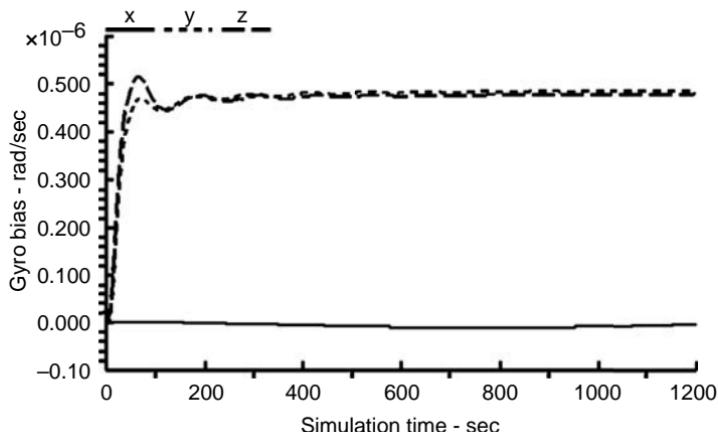
**Fig. 15.7 Inertial tilt angle form estimated gyro biases.**



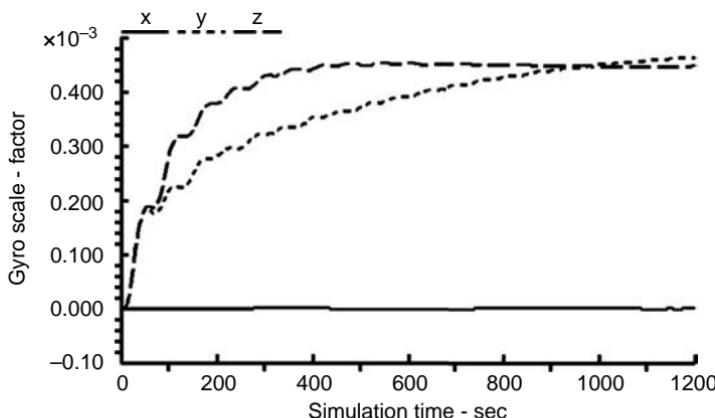
**Fig. 15.8 Inertial tilt angle form estimated gyro scale-factor errors.**

In both cases, a stellar reference point is located along the craft's  $x$  axis for a single vector observation. Cyclical motion about all three axes is imposed on the craft to exercise gyro scale-factor errors. The strap-down gyro's errors are assumed to include bias and scale-factor (see Fig. 6.6). The same error value, whether bias or scale factor, is assumed to exist in each of the three orthogonal sensors. Also included in the filter state vector is a single star sensor's misalignment [Eq. (15.41)].

Shown in Figs. 15.7 and 15.8 are the gyro bias and scale-factor estimates using the inertial attitude error vector model and in Figs. 15.9 and 15.10 are the corresponding results using the reduced-state body quaternion error model. These results are the same and converge to the true error value in the simulation. One



**Fig. 15.9 Body reduced-state quaternion form estimated gyro biases.**



**Fig. 15.10 Body reduced-state quaternion form estimated gyro scale-factor errors.**

interesting feature seen in these results is that, with the observations chosen, the  $x$ -axis bias and scale-factor errors cannot be estimated. This implies that other calibration dynamics and/or additional observation sensors are needed to estimate these errors.

#### 15.4 Chapter Summary

In this chapter, attitude-determination was introduced via an example in terrestrial navigation using low-cost sensors. Several iterative algorithms for determining attitude were presented and compared. Error models for use in a Kalman filter for attitude estimation were developed and compared.



# 16

## Summary

The goal of this book is to provide the reader, whether involved with design, integration, or test and evaluation, with necessary mathematical tools that can be used to evaluate an integrated navigation system's design. Many design choices are illustrated, with each to be considered in light of the navigation system's requirements.

The mathematical techniques presented in this book are grouped into two parts. Part 1 includes coordinate frame definitions, kinematic equations describing motion over a ellipsoidal Earth yielding the system navigation equations, error models for navigation equations, sensor and navigation aid error models, and Kalman filter algorithms for optimally combining navigation systems and aids into an integrated navigation system. In Part 2, application case studies are presented, including inertial sensor laboratory calibration, flight-test evaluation methods, ground alignment, integrated GPS/inertial, in-motion alignment, integrated dead-reckoning/differential global positioning system (DGPS), and attitude determination and estimation for terrestrial and spacecraft navigation systems.

Illustrated and summarized in Parts 1 and 2 are selected navigation implementation reference frames and navigation aid frames. The frames listed are just a few of those in current use and are not all-inclusive. Mathematical methods presented for relating navigation aid data to navigation system implementation's data in error model form can be applied to frames of reference other than those listed.

Examples are presented for modeling navigation system errors. Different methods are illustrated for establishing these error models. Design choices, assumptions, and other integration considerations are required to establish errors and corresponding models used in integrated navigation systems. Some of these considerations for navigation and aids-to-navigation references frames are summarized here:

### *Terrestrial navigation frames*

- 1) Earth-centered
  - a) Inertial
  - b) Earth-fixed
- 2) Local level
  - a) Geographic—north referenced azimuth
  - b) Wander azimuth

*Aids-to-navigation reference frames*

- 1) TACAN
    - a) Range
    - b) Bearing—magnetic north
  - 2) GPS
    - a) Latitude/longitude/altitude/ $v_e/v_n/v_u$ —geographic
    - b)  $x^e/y^e/z^e/x^e/y^e/z^e$ —Earth-centered Earth-fixed
    - c) Pseudorange/delta-range—line-of-sight
  - 3) Doppler
    - a)  $v_x^b/v_y^b/v_z^b$ —body
    - b)  $v_h/v_d/v_v$ —heading referenced
  - 4) Airborne line-of-sight sensors
    - a) Ground mapping radar—relative north and east position
    - b) FLIR—line-of-sight
- Etc.

For a terrestrial GPS/inertial navigation system using one of the possible navigation reference frames just listed, the wander-azimuth frame, choices of implementation options are presented here:

*Wander-Azimuth*

- 1) Position error reference
  - a) Perturbation (true) frame
  - b) Computer frame
- 2) System error states
  - a) Position/velocity/attitude
  - b) Position/position-rate/attitude
- 3) Position Error States
  - a) Angular
  - b) Linear
- 4) Vertical Error
  - a) Separate with horizontal position/velocity or position/position-rate
  - b) Integrated three axis position/velocity or position/position-rate
- 5) Vertical Velocity
  - a)  $v_z = 0$  or  $v_z \neq 0$
- 6) Attitude Error States
  - a) “phi” or “psi” with  $\phi_z = 0$  or  $\delta\theta_z = 0$  &  $\delta\rho_z \neq 0$
  - b) “phi” or “psi” with  $\phi_z \neq 0$  or  $\delta\theta_z \neq 0$  &  $\delta\rho_z = 0$
- 7) Instrument Errors
  - a) Reference frame
    - i. Body- strapdown
    - ii. Navigation- gimbaled
  - b) Stability
    - i. Long term- bias states
    - ii. Short term- process noise
- 8) Measurement Errors
  - a) Pseudo-range: user/satellite clocks, iono, ephemeris
  - b) Horizontal position

- 9) Environmental Errors
  - a) Multi-path
  - b) Gravity

Etc.

Included are the following design choices: position error reference; velocity or position rate-of-change error states; linear or angular position errors; whether to include the vertical axis position and velocity error states; local tilt  $\phi$  or total  $\psi$  attitude error states; body or navigation-frame sensor errors for strap-down or gimballed sensors, respectively; navigation sensors characterized by short- and/or long-term errors; GPS data availability in refined form or line-of-sight single satellite based; and environmental errors' contributions to the error model.

The reference frame and design considerations listed reinforce the notion that all integrated navigation systems are not the same. For example, there are many terrestrial navigation systems' state error dynamic model forms presented in Chapter 5. Insights into which approach is better for a specific application can be gained by evaluations, using simulations and/or actual data. The important work of Huddle [32] provides insights into the qualities of different error model forms. Comparisons by the author [26 and 33] offer additional insights.

Once established, a system design's use of a particular navigation frame, error model, and/or other design consideration must be evaluated based on the intended use of the navigation system. The mathematical methods presented in this book can be used for this purpose and provide the reader, whether involved with the system's design, integration, or test and evaluation, with the tools necessary to perform these evaluations.



## Appendix A

### Pinson Error Model

The navigation system error model developed in this appendix is from Pinson [34 and 35]. Its development shown next is an attempt to enhance the understanding and meaning of the terms in this model consistent with notation used in the other models presented earlier in this book.

#### A.1 Navigation System Error Model

The navigation system error model is based on a nonlocal computational frame  $k$  representation and is expressed in the  $\psi$  form for attitude errors. Later, it will be assumed that  $\phi_z = 0$ .

##### A.1.1 Position Error and Position Error Rate

The Earth-centered inertial frame  $i$  position vector is related to a position defined in the computational frame  $k$  as

$$\mathbf{r}^i = C_k^i \mathbf{r}^k \quad (\text{A.1})$$

The derivative of this equation with respect to time is

$$\begin{aligned} \dot{\mathbf{r}}^i &= C_k^i \dot{\mathbf{r}}^k + \dot{C}_k^i \mathbf{r}^k \\ &= C_k^i \left( \dot{\mathbf{r}}^k + \Omega_{i/k}^k \mathbf{r}^k \right) \end{aligned} \quad (\text{A.2})$$

Taking the derivative again yields

$$\begin{aligned} \ddot{\mathbf{r}}^i &= C_k^i \left( \ddot{\mathbf{r}}^k + \dot{\Omega}_{i/k}^k \mathbf{r}^k + \Omega_{i/k}^k \dot{\mathbf{r}}^k \right) + \dot{C}_k^i \left( \dot{\mathbf{r}}^k + \Omega_{i/k}^k \mathbf{r}^k \right) \\ &= C_k^i \left[ \ddot{\mathbf{r}}^k + 2\Omega_{i/k}^k \dot{\mathbf{r}}^k + \left( \dot{\Omega}_{i/k}^k + \Omega_{i/k}^k \Omega_{i/k}^k \right) \mathbf{r}^k \right] \end{aligned} \quad (\text{A.3})$$

The inertial acceleration separated into specific force and gravitational acceleration is

$$C_i^k \ddot{\mathbf{r}}^i = \mathbf{f}^k + \mathbf{G}^k \quad (\text{A.4})$$

Equating Eq. (A.3) and the transformation of Eq. (A.4) to the computational frame results in

$$\ddot{\mathbf{r}}^k + 2\Omega_{i/k}^k \dot{\mathbf{r}}^k + \left( \dot{\Omega}_{i/k}^k + \Omega_{i/k}^k \Omega_{i/k}^k \right) \mathbf{r}^k = \mathbf{f}^k + \mathbf{G}^k \quad (\text{A.5})$$

This equation is assumed to represent both true dynamics and dynamics as computed within the navigation system's computer.

Equation (A.5) is linearized by assuming the computed values of position and angular rate are equal to the true value plus an error as (computed values are designated with an overbar)

$$\bar{\mathbf{r}}^k = \mathbf{r}^k + \delta\mathbf{r}^k \quad (\text{A.6})$$

$$\bar{\Omega}_{i/k}^k = \Omega_{i/k}^k + \delta\Omega_{i/k}^k \quad (\text{A.7})$$

Substituting Eqs. (A.6) and (A.7) into Eq. (A.5) and ignoring products of error variables result in the following:

$$\begin{aligned} \delta\dot{\mathbf{r}}^k + 2\Omega_{i/k}^k \delta\dot{\mathbf{r}}^k + \left( \dot{\Omega}_{i/k}^k + \Omega_{i/k}^k \Omega_{i/k}^k \right) \delta\mathbf{r}^k &= \delta\mathbf{f}^k + \delta\mathbf{G}^k \\ - 2\delta\Omega_{i/k}^k \dot{\mathbf{r}}^k - \left( \delta\dot{\Omega}_{i/k}^k + \delta\Omega_{i/k}^k \Omega_{i/k}^k + \Omega_{i/k}^k \delta\Omega_{i/k}^k \right) \mathbf{r}^k & \end{aligned} \quad (\text{A.8})$$

The specific force error term in Eq. (A.8) includes misalignment and instrument error and is expanded as

$$\begin{aligned} \delta\mathbf{f}^k &\leftarrow -C_n^k (\boldsymbol{\psi}^n \times) C_k^n \mathbf{f}^k + \delta\mathbf{f}^k \\ &= -(\boldsymbol{\psi}^k \times) \mathbf{f}^k + \delta\mathbf{f}^k \end{aligned} \quad (\text{A.9})$$

Gravitational error is approximated by

$$\delta\mathbf{G}^k \approx \omega_s^2 \left[ \frac{3\mathbf{r}^k \delta h}{R} - \delta\mathbf{r}^k \right] + \delta\mathbf{g}^k \quad (\text{A.10})$$

where

$$\delta\mathbf{g}^k = -C_g^k \Delta\mathbf{G}^g \quad (\text{A.11})$$

$\omega_s$  = Schuler frequency

$R$  = magnitude of Earth's radius

$\delta h$  = altitude error

$\Delta\mathbf{G}^G$  = geographic frame gravity deflections from vertical

Finally, the assumption of a nonlocal computational frame (see earlier alternate velocity error representation developments in Chapter 5) implies the following:

$$\delta\Omega_{i/k}^k \equiv [0] \quad (\text{A.12})$$

and

$$\delta\Omega_{e/k}^k \equiv [0] \quad (\text{A.13})$$

Substituting Eqs. (A.9–A.13) into Eq. (A.8) yields the following:

$$\begin{aligned} \delta\ddot{\mathbf{r}}^k + 2\Omega_{i/k}^k \delta\dot{\mathbf{r}}^k + & \left( \dot{\Omega}_{i/k}^k + \Omega_{i/k}^k \Omega_{i/k}^k + \omega_s^2 I \right) \delta\mathbf{r}^k \\ = -(\boldsymbol{\psi}^k \times) \mathbf{f}^k + \delta\mathbf{f}^k + 3\frac{\omega_s^2}{R} \mathbf{r}^k \delta h + \delta\mathbf{g}^k \end{aligned} \quad (\text{A.14})$$

In the following, Eq. (A.14) will be expanded and expressed in component form. Each term in the resulting component form will be evaluated by examining its relative contribution to the dynamics model. Only significant terms will be retained in the finalized dynamics model.

The first term in the parentheses in Eq. (A.14) is expanded as (using the angular velocity addition theorem)

$$\Omega_{i/k}^k = \Omega_{i/e}^k + \Omega_{e/k}^k \quad (\text{A.15})$$

or, in vector form,

$$\boldsymbol{\omega}_{i/k}^k = \boldsymbol{\Omega} + \boldsymbol{\rho} \quad (\text{A.16})$$

The first term on the right-hand side of Eq. (A.15) can be expressed as (using the similarity transformation)

$$\Omega_{i/e}^k = C_e^k \Omega_{i/e}^e C_k^e \quad (\text{A.17})$$

Taking the derivative of this expression results in

$$\begin{aligned} \dot{\Omega}_{i/e}^k &= \dot{C}_e^k \Omega_{i/e}^e C_k^e + C_e^k \dot{\Omega}_{i/e}^e C_k^e + C_e^k \Omega_{i/e}^e \dot{C}_k^e \\ &= -\Omega_{e/k}^k C_e^k \Omega_{i/e}^e C_k^e + C_e^k \Omega_{i/e}^e C_k^e \Omega_{e/k}^k \\ &= -\Omega_{e/k}^k \Omega_{i/e}^k + \Omega_{i/e}^k \Omega_{e/k}^k \end{aligned} \quad (\text{A.18})$$

where the following relationships have been used:

$$\dot{\Omega}_{i/e}^e \equiv [0] \quad (\text{A.19})$$

$$\dot{C}_e^k = -\Omega_{e/k}^k C_e^k \quad (\text{A.20})$$

and

$$\dot{C}_k^e = C_k^e \Omega_{e/k}^k \quad (\text{A.21})$$

Expanding Eq. (A.18) using components of vectors defined in Eq. (A.16) results in

$$\dot{\Omega}_{i/k}^k = \begin{bmatrix} 0 & -\dot{\rho}_z + \rho_x \Omega_y - \rho_y \Omega_x & \dot{\rho}_y + \rho_x \Omega_z - \rho_z \Omega_x \\ \dot{\rho}_z - \rho_x \Omega_y + \rho_y \Omega_x & 0 & -\dot{\rho}_x + \rho_y \Omega_z - \rho_z \Omega_y \\ -\dot{\rho}_y - \rho_x \Omega_z + \rho_z \Omega_x & \dot{\rho}_x - \rho_y \Omega_z + \rho_z \Omega_y & 0 \end{bmatrix} \quad (\text{A.22})$$

The second term in the parentheses in Eq. (A.14) is expressed as

$$\Omega_{i/k}^k \Omega_{i/k}^k = \begin{bmatrix} -\omega_y^2 - \omega_z^2 & \omega_x \omega_y & \omega_x \omega_z \\ \omega_y \omega_x & -\omega_x^2 - \omega_z^2 & \omega_y \omega_z \\ \omega_z \omega_x & \omega_z \omega_y & -\omega_x^2 - \omega_y^2 \end{bmatrix} \quad (\text{A.23})$$

The matrices in Eqs. (A.22) and (A.23) are added as indicated in Eq. (A.14). Resulting matrix components can be simplified by using the following:

$$\omega^2 = \boldsymbol{\omega} \cdot \boldsymbol{\omega} = \omega_x^2 + \omega_y^2 + \omega_z^2 \quad (\text{A.24})$$

$$\Omega^2 = \boldsymbol{\Omega} \cdot \boldsymbol{\Omega} = \Omega_x^2 + \Omega_y^2 + \Omega_z^2 \quad (\text{A.25})$$

The three terms in the first row of the matrix, resulting from adding Eqs. (A.22) and (A.23), will be illustrated next. The (1,1) element sum is

$$\begin{aligned} -\omega_y^2 - \omega_z^2 &= \omega_x^2 - \omega^2 \\ &= (\rho_x + \Omega_x)(\rho_x + \Omega_x) - \omega^2 \\ &= \Omega_x^2 + \rho_x(\rho_x + 2\Omega_x) - \omega^2 \\ &= \Omega_x^2 + \rho_x(\omega_x + \Omega_x) - \omega^2 \end{aligned} \quad (\text{A.26})$$

The (1,2) element sum is

$$\begin{aligned} -\dot{\rho}_z + \rho_x \Omega_y - \rho_y \Omega_x + \omega_x \omega_y &= -\dot{\rho}_z + \rho_x \Omega_y - \rho_y \Omega_x + (\rho_x + \Omega_x)(\rho_y + \Omega_y) \\ &= -\dot{\rho}_z + \rho_x \Omega_y - \rho_y \Omega_x + \rho_x \rho_y + \rho_x \Omega_y + \rho_y \Omega_x + \Omega_x \Omega_y \\ &= -\dot{\rho}_z + \rho_x \Omega_y + \rho_x \rho_y + \rho_x \Omega_y + \Omega_x \Omega_y \\ &= -\dot{\rho}_z + \rho_x(\rho_y + \Omega_y + \Omega_y) + \Omega_x \Omega_y \\ &= -\dot{\rho}_z + \rho_x(\omega_y + \Omega_y) + \Omega_x \Omega_y \end{aligned} \quad (\text{A.27})$$

The (1,3) element sum is

$$\begin{aligned}
 \dot{\rho}_y + \rho_x \Omega_z - \rho_z \Omega_x + \omega_x \omega_z &= \dot{\rho}_y + \rho_x \Omega_z - \rho_z \Omega_x + (\rho_x + \Omega_x)(\rho_z + \Omega_z) \\
 &= \dot{\rho}_y + \rho_x \Omega_z - \rho_z \Omega_x + \rho_x \rho_z + \rho_x \Omega_z + \rho_z \Omega_x + \Omega_x \Omega_z \\
 &= \dot{\rho}_y + \rho_x \Omega_z + \rho_x \rho_z + \rho_x \Omega_z + \Omega_x \Omega_z \\
 &= \dot{\rho}_y + \rho_x (\rho_z + \Omega_z + \Omega_z) + \Omega_x \Omega_z \\
 &= \dot{\rho}_y + \rho_x (\omega_z + \Omega_z) + \Omega_x \Omega_z
 \end{aligned} \tag{A.28}$$

The other elements of the matrix sum follow in a similar manner. The term in parentheses in Eq. (A.14) becomes

$$() = \begin{bmatrix} \omega_s^2 - \omega^2 + \Omega_x^2 + \rho_x(\omega_x + \Omega_x) & -\dot{\rho}_z + \rho_x(\omega_y + \Omega_y) + \Omega_x \Omega_y & \dot{\rho}_y + \rho_x(\omega_z + \Omega_z) + \Omega_x \Omega_z \\ \dot{\rho}_z + \rho_y(\omega_x + \Omega_x) + \Omega_y \Omega_x & \omega_s^2 - \omega^2 + \Omega_y^2 + \rho_y(\omega_y + \Omega_y) & -\dot{\rho}_x + \rho_y(\omega_z + \Omega_z) + \Omega_y \Omega_z \\ -\dot{\rho}_y + \rho_z(\omega_x + \Omega_x) + \Omega_z \Omega_x & \dot{\rho}_x + \rho_z(\omega_y + \Omega_y) + \Omega_z \Omega_y & \omega_s^2 - \omega^2 + \Omega_z^2 + \rho_z(\omega_z + \Omega_z) \end{bmatrix} \tag{A.29}$$

The expansion for other terms in Eq. (A.14) is straightforward, and results are shown in the error model summary to follow.

### A.1.2 “Psi” Attitude Error

The dynamics for the  $\psi$  form of the attitude error equations is developed in the following. The body-to-computational frame direction cosine matrix dynamics is given by

$$\begin{aligned}
 \dot{C}_b^k &= C_b^k \Omega_{k/b}^b \\
 &= C_b^k (\Omega_{i/b}^b - \Omega_{i/k}^b)
 \end{aligned} \tag{A.30}$$

This matrix differential equation is assumed to represent both true and computed variables. The computed direction cosine matrix and rates are also represented as

$$\bar{C}_b^k = [I - (\boldsymbol{\psi}^k \times)] C_b^k \tag{A.31}$$

$$\bar{\Omega}_{i/b}^b = \Omega_{i/b}^b + \delta \Omega_{i/b}^b \tag{A.32}$$

$$\bar{\Omega}_{i/k}^b = \Omega_{i/k}^b + \delta \Omega_{i/k}^b \tag{A.33}$$

Taking the derivative of Eq. (A.31) yields

$$\bar{\dot{C}}_b^k = \dot{C}_b^k - (\dot{\boldsymbol{\psi}}^k \times) C_b^k - (\boldsymbol{\psi}^k \times) \dot{C}_b^k \tag{A.34}$$

Expressing Eq. (A.30) as the computed matrix results in

$$\bar{C}_b^k = \bar{C}_b^k \left( \bar{\Omega}_{i/b}^b - \bar{\Omega}_{i/k}^b \right) \quad (\text{A.35})$$

Equating the right-hand sides of Eqs. (A.34) and (A.35) and using Eqs. (A.32) and (A.33) result in the following matrix differential equation:

$$\begin{aligned} (\dot{\psi}^k \times) &= -C_b^k \left( \delta\Omega_{i/b}^b - \delta\Omega_{i/k}^b \right) C_k^b \\ &= -C_b^k \delta\Omega_{i/b}^b C_k^b + C_b^k \delta\Omega_{i/k}^b C_k^b \end{aligned} \quad (\text{A.36})$$

or, in vector form,

$$\dot{\psi}^k = -C_b^k \delta\omega_{i/b}^b + C_b^k \delta\omega_{i/k}^b \quad (\text{A.37})$$

The error in computed angular rate of the computational frame relative to the inertial frame and referenced in the body frame  $\delta\omega_{i/k}^b$  is obtained from the following:

$$\begin{aligned} \bar{\omega}_{i/k}^b &= \bar{C}_k^b \bar{\omega}_{i/k}^k \\ &= C_k^b [I + (\psi^k \times)] \left( \omega_{i/k}^k + \delta\omega_{i/k}^k \right) \\ &\approx C_k^b \omega_{i/k}^k + C_k^b \delta\omega_{i/k}^k + C_k^b (\psi^k \times) \omega_{i/k}^k \end{aligned} \quad (\text{A.38})$$

or

$$\begin{aligned} C_b^k \delta\omega_{i/k}^b &= \delta\omega_{i/k}^k + (\psi^k \times) \omega_{i/k}^k \\ &= \delta\omega_{i/k}^k - \Omega_{i/k}^k \psi^k \end{aligned} \quad (\text{A.39})$$

Substituting Eq. (A.39) into Eq. (A.37) yields

$$\dot{\psi}^k + \Omega_{i/k}^k \psi^k = -\delta\omega_{i/b}^k + \delta\omega_{i/k}^k \quad (\text{A.40})$$

Or, because the nonlocal computational frame implies the relationship in Eq. (A.12), attitude error dynamics become

$$\dot{\psi}^k + \Omega_{i/k}^k \psi^k = -\delta\omega_{i/b}^k \quad (\text{A.41})$$

The last term in Eq. (A.41) represents gyro drift referenced in the computational frame.

### A.1.3 System Error Dynamic Model Summary

Error dynamics are summarized from Eqs. (A.14) and (A.41) in Figure A.1. Elements of this preceding matrix can be simplified by examining their magnitudes. Consider the magnitudes of Earth rate  $\Omega$ , transport rate  $\rho$ , and

$$\frac{d}{dt} \begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta r_z \\ \delta \Omega_x \\ \delta \Omega_y \\ \delta \Omega_z \\ \Psi_x \\ \Psi_y \\ \Psi_z \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\omega_s^2 - \Omega_x^2 - \Omega_z^2 - \rho_x(\Omega_x + \Omega_z) & \rho_z - \rho_x(\Omega_y + \Omega_z) - \Omega_x \Omega_y & -\rho_y - \rho_x(\Omega_z + \Omega_z) - \Omega_x \Omega_z & 0 & 2\Omega_z & -2\Omega_y & 0 & -f_z & f_y \\ -\rho_z - \rho_y(\Omega_x + \Omega_z) - \Omega_y \Omega_z & -\omega_s^2 - \omega^2 - \Omega_y^2 - \rho_y(\Omega_y + \Omega_z) & \rho_x - \rho_y(\Omega_z + \Omega_z) - \Omega_y \Omega_z & -2\Omega_z & 0 & 2\Omega_x & f_z & 0 & -f_x \\ \rho_y - \rho_z(\Omega_x + \Omega_z) - \Omega_z \Omega_x & -\rho_x - \rho_z(\Omega_y + \Omega_z) - \Omega_z \Omega_y & -\omega_s^2 - \omega^2 - \Omega_z^2 - \rho_z(\Omega_z + \Omega_z) & 2\Omega_y & -2\Omega_x & 0 & -f_y & f_x & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\omega_z & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \omega_x & -\omega_x \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \omega_y & 0 \end{bmatrix}$$

$$\times \begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta r_z \\ \delta \Omega_x \\ \delta \Omega_y \\ \delta \Omega_z \\ \Psi_x \\ \Psi_y \\ \Psi_z \end{bmatrix} + \begin{bmatrix} \delta g_x \\ \delta g_y \\ \delta g_z \\ \epsilon_x \\ \epsilon_y \\ \epsilon_z \end{bmatrix}$$

Fig. A.1 System error dynamic model.

the Schuler rate  $\omega_s$ .

$$\Omega^2 \sim 5.31 \times 10^{-9} \quad (\text{rad/s})^2 \quad (\text{A.42})$$

$$|\rho| \leq |\Omega| \quad \text{subsonic aircraft} \quad (\text{A.43})$$

$$\omega_s^2 = 1.54 \times 10^{-6} \quad (\text{rad/s})^2 \quad (\text{A.44})$$

The other terms in elements (4,1), (5,2), and (6,3) are much smaller than  $\omega_s^2$  and can be neglected. Other terms, for example, (4,2) and (5,1), are smaller than the (4,1) and (5,2) terms, so that it may be assumed, for equal position error magnitudes, that (4,1) and (5,2) elements govern the coupling of position errors into their corresponding components. For quasi-steady motion, transport rates are assumed negligible.

The terms that govern the rate of change of position rate  $\delta\dot{r}$  are evaluated based on expected magnitudes of the navigation system errors as follows:

$$\omega_s^2 \delta r \sim 1.54 \times 10^{-6} (4860 \text{ ft}) = 0.007 \text{ ft/s}^2 \quad (\text{A.45})$$

$$\Omega \delta \dot{r} \sim 7.29 \times 10^{-5} (2.5 \text{ ft/s}) = 0.002 \text{ ft/s}^2 \quad (\text{A.46})$$

$$f_z \phi \sim 32.2 (0.00087 \text{ rad}) = 0.028 \text{ ft/s}^2 \quad (\text{A.47})$$

Although the position error rate term in Eq. (A.46) is much smaller than the attitude term in Eq. (A.47), it is comparable to the position error magnitude in Eq. (A.45). This term also provides cross-axis couplings. Therefore, this position rate error term is retained.

The navigation system error model, including only the horizontal position components, using the preceding numerical arguments, and referring back to Eqs. (A.14) and (A.41), simplifies to

$$\begin{aligned} \frac{d}{dt} \begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta \dot{r}_x \\ \delta \dot{r}_y \\ \psi_x \\ \psi_y \\ \psi_z \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ -\omega_s^2 & 0 & 0 & 2\Omega_z & 0 & -f_z & f_y \\ 0 & -\omega_s^2 & -2\Omega_z & 0 & f_z & 0 & -f_x \\ 0 & 0 & 0 & 0 & 0 & \omega_z & -\omega_y \\ 0 & 0 & 0 & 0 & -\omega_z & 0 & \omega_x \\ 0 & 0 & 0 & 0 & \omega_y & -\omega_x & 0 \end{bmatrix} \begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta \dot{r}_x \\ \delta \dot{r}_y \\ \psi_x \\ \psi_y \\ \psi_z \end{bmatrix} \\ &+ \begin{bmatrix} 0 \\ 0 \\ \delta f_x \\ \delta f_y \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \delta g_x \\ \delta g_y \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \end{bmatrix} \quad (\text{A.48}) \end{aligned}$$

Additional sensor errors in the bottom rows in Eq. (A.48) include both long-term errors, that is, biases, and short-term errors, that is, gyro random walk. Long-term errors, gyro and accelerometer biases, are included as additional error states adjoined to those just given. Short-term errors, gravity disturbances and gyro random walk, should be included as process noise and included in the filter's  $Q$  matrix.

#### A.1.4 Navigation Solution Corrections

Computational frame estimates of position and position rate and corresponding attitudes are reformed as corrections to the navigation solution. Velocity corrections are obtained from combinations of navigation filter states. The navigation frame velocity can be expressed as

$$\mathbf{v}^n = C_k^n \left( \dot{\mathbf{r}}^k + \Omega_{e/k}^k \mathbf{r}^k \right) \quad (\text{A.49})$$

This equation is assumed to represent true and computed values. Taking the variation of Eq. (A.49), the velocity error becomes

$$\begin{aligned} \delta\mathbf{v}^n &= \delta C_k^n \left( \dot{\mathbf{r}}^k + \Omega_{e/k}^k \mathbf{r}^k \right) + C_k^n \left( \delta \dot{\mathbf{r}}^k + \delta \Omega_{e/k}^k \mathbf{r}^k + \Omega_{e/k}^k \delta \mathbf{r}^k \right) \\ &= \delta C_k^n C_n^k \mathbf{v}^n + C_k^n \left( \delta \dot{\mathbf{r}}^k + \delta \Omega_{e/k}^k \mathbf{r}^k + \Omega_{e/k}^k \delta \mathbf{r}^k \right) \\ &= -(\delta \boldsymbol{\theta} \times) C_k^n C_n^k \mathbf{v}^n + C_k^n \left( \delta \dot{\mathbf{r}}^k + \Omega_{e/k}^k \delta \mathbf{r}^k \right) \\ &= -(\delta \boldsymbol{\theta} \times) \mathbf{v}^n + C_k^n \left( \delta \dot{\mathbf{r}}^k + \Omega_{e/k}^k \delta \mathbf{r}^k \right) \end{aligned} \quad (\text{A.50})$$

where Eq. (A.13) and the following relationship are used in Eq. (A.50):

$$\delta C_k^n = -(\delta \boldsymbol{\theta} \times) C_k^n \quad (\text{A.51})$$

Equation (A.50) is expanded and rewritten for the two horizontal velocity components as

$$\begin{bmatrix} \delta v_x \\ \delta v_y \\ - \end{bmatrix} = \begin{bmatrix} 0 & \delta \theta_z & -\delta \theta_y \\ -\delta \theta_z & 0 & \delta \theta_x \\ - & - & - \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} + C_k^n \left\{ \begin{bmatrix} \delta \dot{r}_x \\ \delta \dot{r}_y \\ - \end{bmatrix} + \begin{bmatrix} 0 & -\rho_z & \rho_y \\ \rho_z & 0 & -\rho_x \\ - & - & - \end{bmatrix} \begin{bmatrix} \delta r_x \\ \delta r_y \\ \delta r_z \end{bmatrix} \right\} \quad (\text{A.52})$$

Assuming that  $v_z$  is small, such that its product with error states is negligible and  $\rho_z \equiv 0$ , then the horizontal velocity errors can be written in component form as

$$\delta v_x \approx \delta \dot{r}_x + v_y \delta \theta_z \quad (\text{A.53})$$

$$\delta v_y \approx \delta \dot{r}_y - v_x \delta \theta_z \quad (\text{A.54})$$

The assumption is that  $\phi_z = 0$  results in the following relationship:

$$\delta\theta_z = -\psi_z \quad (\text{A.55})$$

Equations (A.53) and (A.54) then become

$$\delta v_x \approx \delta \dot{r}_x - v_y \psi_z \quad (\text{A.56})$$

$$\delta v_y \approx \delta \dot{r}_y + v_x \psi_z \quad (\text{A.57})$$

Horizontal tilt errors are obtained from the states by

$$\phi_x = \psi_x + \delta\theta_x \quad (\text{A.58})$$

$$\phi_y = \psi_y + \delta\theta_y \quad (\text{A.59})$$

$$\phi_z = 0 \quad (\text{A.60})$$

where

$$\delta\theta_x = -\frac{\delta r_y}{R} \quad (\text{A.61})$$

$$\delta\theta_y = \frac{\delta r_x}{R} \quad (\text{A.62})$$

Using these delta theta values, corrections for  $C_n^e$  matrix elements can be obtained by

$$\delta C_e^n = -(\delta\theta \times) C_e^n \quad (\text{A.63})$$

## A.2 Global-Positioning-System Latitude and Longitude Position as Measurements

The global positioning system (GPS) position is referenced to a local geodetic frame  $g$ , that is, latitude/longitude. Measurements are processed in the navigation filter, which is formulated in the computational frame  $k$ . Using these GPS data as measurements, it is required to transform these local geodetic frame data into this computational frame. Consider the following position errors formed from the difference in GPS and inertial navigation unit (INU) latitude  $\phi$  and longitude  $\lambda$ :

$$\Delta r_n = R(\phi_{\text{GPS}} - \phi_{\text{INU}}) = R\Delta\phi \quad (\text{A.64})$$

$$\Delta r_w = R(\lambda_{\text{GPS}} - \lambda_{\text{INU}}) \cos \phi = R\Delta\lambda c\phi \quad (\text{A.65})$$

Geodetic frame errors can be transformed into the navigation frame by

$$\begin{aligned} \Delta \bar{r}^k &= \bar{C}_g^k \Delta r^g \\ &= \begin{bmatrix} c\bar{\alpha} & s\bar{\alpha} \\ -s\bar{\alpha} & c\bar{\alpha} \end{bmatrix} \begin{bmatrix} \Delta r_n \\ \Delta r_w \end{bmatrix} \end{aligned} \quad (\text{A.66})$$

where the wander angle  $\bar{\alpha}$  is obtained from the INU. This angle is in error and is represented as

$$\bar{\alpha} = \alpha + \delta\alpha \quad (\text{A.67})$$

Substituting this expression into Eq. (A.66) and ignoring products of error variables yield

$$\begin{aligned} \begin{bmatrix} \Delta\bar{r}_x \\ \Delta\bar{r}_y \end{bmatrix} &= \begin{bmatrix} c(\alpha + \delta\alpha)\Delta r_n + s(\alpha + \delta\alpha)\Delta r_w \\ -s(\alpha + \delta\alpha)\Delta r_n + c(\alpha + \delta\alpha)\Delta r_w \end{bmatrix} \\ &= \begin{bmatrix} \delta r_x \\ \delta r_y \end{bmatrix} + \begin{bmatrix} -s\alpha\Delta r_n + c\alpha\Delta r_w \\ -c\alpha\Delta r_n - s\alpha\Delta r_w \end{bmatrix}\delta\alpha \\ &= \begin{bmatrix} \delta r_x \\ \delta r_y \end{bmatrix} + \begin{bmatrix} \Delta r_y \\ -\Delta r_x \end{bmatrix}\delta\alpha \end{aligned} \quad (\text{A.68})$$

The wander angle error can be expressed as

$$\begin{aligned} \delta\alpha &= \delta\theta_z - \delta\lambda \sin\phi \\ &= \delta\theta_z + \frac{t\phi}{R} [s\alpha \quad c\alpha] \begin{bmatrix} \delta r_x \\ \delta r_y \end{bmatrix} \end{aligned} \quad (\text{A.69})$$

which, when substituted into Eq. (A.68), yields

$$\begin{bmatrix} \Delta\bar{r}_x \\ \Delta\bar{r}_y \end{bmatrix} = \begin{bmatrix} 1 + \frac{\Delta r_y t \phi s\alpha}{R} & \frac{\Delta r_y t \phi c\alpha}{R} \\ -\frac{\Delta r_x t \phi s\alpha}{R} & 1 - \frac{\Delta r_x t \phi c\alpha}{R} \end{bmatrix} \begin{bmatrix} \delta r_x \\ \delta r_y \end{bmatrix} + \begin{bmatrix} \Delta r_y \\ -\Delta r_x \end{bmatrix} \delta\theta_z \quad (\text{A.70})$$

An approximation to this equation is

$$\begin{bmatrix} \Delta\bar{r}_x \\ \Delta\bar{r}_y \end{bmatrix} \approx \begin{bmatrix} \delta r_x \\ \delta r_y \end{bmatrix} + \begin{bmatrix} \Delta r_y \\ -\Delta r_x \end{bmatrix} \delta\theta_z \quad (\text{A.71})$$

or, from Eq. (A.55),

$$\begin{bmatrix} \Delta\bar{r}_x \\ \Delta\bar{r}_y \end{bmatrix} \approx \begin{bmatrix} \delta r_x \\ \delta r_y \end{bmatrix} + \begin{bmatrix} -\Delta r_y \\ \Delta r_x \end{bmatrix} \psi_z \quad (\text{A.72})$$

Expressing Eq. (A.72) in terms of a linear measurement matrix using the state vector just defined yields the following:

$$H_{\Delta r} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -\Delta r_y & \cdots \\ 0 & 1 & 0 & 0 & 0 & 0 & \Delta r_x & \cdots \end{bmatrix} \quad (\text{A.73})$$



## Appendix B Orbital Dynamics

### B.1 Orbital Position in Earth-Centered Frame

The solution to central force trajectory equations in Chapter 7

$$\ddot{r} - r\dot{\theta}^2 = -\frac{\mu}{r^2} \quad (7.20)$$

$$r\ddot{\theta} + 2\dot{r}\dot{\theta} = 0 \quad (7.21)$$

is outlined next.

An immediate result can be obtained from the second of these equations. The differential of the following product is related to this equation as

$$\begin{aligned} \frac{d}{dt}(r^2\dot{\theta}) &= 2r\dot{r}\dot{\theta} + r^2\ddot{\theta} \\ &= r(2\dot{r}\dot{\theta} + r\ddot{\theta}) \Rightarrow r^2\dot{\theta} \equiv h - \text{const} \\ &= 0 \end{aligned} \quad (\text{B.1})$$

The second of these relationships is zero by Eq. (7.21), yielding the result that the trajectory angular momentum  $h$  is a constant. From this result, another result of importance is obtained. The “area” velocity is the area enclosed per unit of time. From Fig. 7.6, the differential area covered by the position change is

$$dA = \frac{1}{2}(rd\theta)r \quad (\text{B.2})$$

Dividing this equation by  $dt$ , the area velocity becomes

$$\dot{A} = \frac{1}{2}r^2\dot{\theta} = \frac{1}{2}h - \text{const} \quad (\text{B.3})$$

This result states that the area velocity is also a constant, which is Kepler’s Second Law.

The first of the trajectory equations is now addressed. The rate of change of trajectory's radius can be expressed as the following sequence (see [36]):

$$\dot{r} = \frac{dr}{dt} = \frac{dr}{d\theta} \frac{d\theta}{dt} = \frac{dr}{d\theta} \dot{\theta} = \frac{h}{r^2} \frac{dr}{d\theta} = -h \frac{d}{d\theta} \left( \frac{1}{r} \right) \quad (\text{B.4})$$

With the suggested change of variables in this equation  $r = 1/u$ , the radius rate of change becomes

$$\dot{r} = -h \frac{du}{d\theta} \quad (\text{B.5})$$

Continuing, the second derivative becomes

$$\ddot{r} = -h \frac{d}{dt} \frac{du}{d\theta} = -h \frac{d}{d\theta} \frac{d\theta}{dt} \frac{du}{d\theta} = -h \frac{d^2 u}{d\theta^2} \dot{\theta} = -hu^2 \frac{d^2 u}{d\theta^2} \quad (\text{B.6})$$

and then substituting these expressions into Eq. (7.20) yields

$$-hu^2 \frac{d^2 u}{d\theta^2} - \frac{1}{u} (hu^2)^2 = -\mu u^2 \quad (\text{B.7})$$

and, dividing through by  $-hu^2$ , this equation expressed in the variable  $u$  becomes

$$\frac{d^2 u}{d\theta^2} + u = \frac{\mu}{h^2} \quad (\text{B.8})$$

This second-order linear nonhomogenous equation has a solution of the form

$$u = \frac{\mu}{h^2} + C \cos(\theta - \omega) \quad (\text{B.9})$$

where  $C$  and  $\omega$  are integration constants. These constants are evaluated at  $\theta = \omega$ , where  $u$  and  $r$  are constants also. Examining the sum of kinetic and potential energy (because this system is conservative, the total energy is a constant) at this point,

$$\begin{aligned} \text{TE} = \text{KE} + \text{PE} &= \frac{1}{2}(r\dot{\theta})^2 - \frac{\mu}{r} \\ &= \frac{1}{2}h^2u^2 - \mu u \end{aligned}$$

at  $\theta = \omega$

$$\begin{aligned} &= \frac{1}{2} h^2 \left( \frac{\mu}{h^2} + C \right)^2 - \mu \left( \frac{\mu}{h^2} + C \right) \\ &\Rightarrow C = \frac{\sqrt{2TE + \mu^2/h^2}}{h} \\ &= \frac{1}{2} h^2 C^2 - \frac{1}{2} \frac{\mu^2}{h^2} \end{aligned} \quad (\text{B.10})$$

Therefore

$$u = \frac{\mu}{h^2} \left\{ 1 + \sqrt{1 + \frac{2TE}{\mu^2} \frac{h^2}{h^2}} \cos(\theta - \omega) \right\} \quad (\text{B.11})$$

or, expressed in terms of the trajectory radial position

$$r = \frac{h^2}{\mu} \left/ \left\{ 1 + \sqrt{1 + \frac{2TE}{\mu^2} \frac{h^2}{h^2}} \cos(\theta - \omega) \right\} \right. \quad (\text{B.12})$$

Referring to Fig. 7.7, the following expression for the radial position is used to obtain a relationship between the orbit's angular momentum, gravitational constant, and the orbit's eccentricity:

$$r = \frac{a(1 - \varepsilon^2)}{1 + \varepsilon \cos f} \Rightarrow a(1 - \varepsilon^2) = \frac{h^2}{\mu} \quad (\text{B.13})$$

Specializing Eq. (B.12) for an elliptical trajectory, the following relationship, based on the geometry of an ellipse, will be used in the following:

$$b = a(1 - \varepsilon^2)^{1/2} \quad (\text{B.14})$$

With this relationship, the expression for the orbital period can be obtained. The orbital period is defined as the area of the ellipse divided by the aerial velocity. From Eqs. (B.14) and (B.3), the orbital period becomes

$$\tau = \frac{\pi ab}{\dot{A}} = \frac{\pi a(1 - \varepsilon^2)^{1/2}}{\frac{1}{2}h} = \frac{2\pi a(1 - \varepsilon^2)^{1/2}}{\mu^{1/2} a^{1/2} (1 - \varepsilon^2)^{1/2}} \quad (\text{B.15})$$

Or

$$\tau = 2\pi \sqrt{\frac{a^3}{\mu}} \quad (\text{B.16})$$

This equation is a statement of Kepler's Third Law.

In the following, a development of Kepler's equation (before Newton) based on geometrical concepts is presented. From the Second Law, the orbital period  $\tau$

sweeping out full elliptical area  $\pi ab$  and a fractional portion of this period  $t$  sweeping out the area  $A_{\text{FP}}$  are related as

$$\frac{t}{\tau} = \frac{A_{\text{FP}}}{\pi ab} \quad (\text{B.17})$$

In Table B.1 is a sequence of areas relating the area covered in the elliptical trajectory to a corresponding area for a circular trajectory, traditionally referred to as the “auxiliary circle.” The elliptical area is related to the circular area by the scaling parameter shown in the last of these sequences. The ratio in Eq. (B.17) becomes

$$\begin{aligned} \frac{t}{\tau} &= \frac{b}{a^2} \frac{1}{2} a^2 (E - \varepsilon \sin E) \frac{1}{\pi ab} \\ &= \frac{1}{2\pi} (E - \varepsilon \sin E) \end{aligned} \quad (\text{B.18})$$

Define the mean anomaly as

$$M \equiv \frac{2\pi t}{\tau} \quad (\text{B.19})$$

Then the following relationship is obtained:

$$M = E - \varepsilon \sin E \quad (\text{B.20})$$

This equation is known as Kepler's equation, relating the mean anomaly to the eccentric anomaly  $E$  defined in Fig. 7.7.

Various relationships between the auxiliary circle variables and the ellipse variables can be obtained

$$\begin{aligned} AP = r \sin f &= \frac{b}{a} AP' = \frac{b}{a} a \sin E \\ &= \frac{a(1 - \varepsilon^2)^{1/2}}{a} a \sin E = a(1 - \varepsilon^2)^{1/2} \sin E \end{aligned} \quad (\text{B.21})$$

$$\frac{a(1 - \varepsilon^2)}{1 + \varepsilon \cos f} \sin f = a(1 - \varepsilon^2)^{1/2} \sin E$$

Or

$$\sin E = \frac{(1 - \varepsilon^2)^{1/2}}{1 + \varepsilon \cos f} \sin f \quad (\text{B.22})$$

Others follow. The distance from the origin to the focus is expressed as

$$ae = a \cos E - r \cos f \quad (\text{B.23})$$

**Table B.1 Area solution for Kepler's equation**

Area	Figure	Equation
I		$\frac{1}{2}a^2E$
II		$\frac{1}{2}a^2 \cos E \sin E$
III		$\frac{1}{2}a^2(\cos E - \varepsilon) \sin E$
IV		$A_{FP'} = A_I - A_{II} + A_{III}$ $\frac{1}{2}a^2(E - \varepsilon \sin E)$
V		$\begin{aligned} A_{FP} &= \frac{b}{a}A_{FP'} \\ &= \frac{a(1 - \varepsilon^2)^{1/2}}{a} \frac{1}{2}a^2(E - \varepsilon \sin E) \\ &= \frac{1}{2}a^2(1 - \varepsilon^2)^{1/2}(E - \varepsilon \sin E) \end{aligned}$ <p>because</p> $AP = \frac{b}{a}AP'$

Then

$$r \cos f = a(\cos E - \varepsilon) \quad (\text{B.24})$$

With this result and Eq. (B.21), the following sequence is used to establish an expression for the orbit radius:

$$\begin{aligned} r^2 \sin^2 f + r^2 \cos^2 f &= r^2 = a^2(1 - \varepsilon^2) \sin^2 E + a^2(\cos E - \varepsilon)^2 \\ &= a^2(\sin^2 - \varepsilon^2 \sin^2 E + \cos^2 E - 2\varepsilon \cos E + \varepsilon^2) \\ &= a^2(1 - \varepsilon^2 \sin^2 E - 2\varepsilon \cos E + \varepsilon^2) \\ &= a^2[1 + \varepsilon^2(1 - \sin^2 E) - 2\varepsilon \cos E] \\ &= a^2(1 + \varepsilon^2 \cos^2 E - 2\varepsilon \cos E) \\ &= a^2(1 - \varepsilon \cos E)^2 \end{aligned} \quad (\text{B.25})$$

Or

$$r = a(1 - \varepsilon \cos E) \quad (\text{B.26})$$

Using this result in Eq. (B.26), with the following sequence,

$$\begin{aligned} a(1 - \varepsilon \cos E) \cos f &= a(\cos E - \varepsilon) \\ (1 - \varepsilon \cos E) \cos f &= (\cos E - \varepsilon) \\ \cos f - \varepsilon \cos E \cos f &= \cos E - \varepsilon \\ \cos E(1 + \varepsilon \cos f) &= \varepsilon + \cos f \end{aligned} \quad (\text{B.27})$$

Or

$$\cos E = \frac{\varepsilon + \cos f}{(1 + \varepsilon \cos f)} \quad (\text{B.28})$$

Rearranging this equation results in the following expression for the true anomaly:

$$\cos f = \frac{\cos E - \varepsilon}{(1 - \varepsilon \cos E)} \quad (\text{B.29})$$

Before leaving Kepler's equation, the same result in Eq. (B.18) can be obtained by direct integration (after Newton). The differential area in terms of the true anomaly is

$$dA = \frac{1}{2}(r d\theta)r = \frac{1}{2}r^2 df \quad (\text{B.30})$$

A change of variable from the true to the eccentric anomaly is needed. This relationship is established in the following sequence by starting with Eq. (B.29):

$$\begin{aligned}
 df &= \frac{\sin E}{\sin f(1 - \varepsilon \cos E)} \frac{(1 - \varepsilon^2)}{(1 - \varepsilon \cos E)} dE \\
 &= \frac{\sin E}{\sin f(1 - \varepsilon \cos E)} (1 + \varepsilon \cos f) dE \\
 \sin f \, df &= \frac{(1 - \varepsilon^2) \sin E}{(1 - \varepsilon \cos E)^2} dE \Rightarrow = \frac{\sin E (1 + \varepsilon \cos f)}{\sin f} \frac{1}{(1 - \varepsilon \cos E)} dE \\
 &= (1 - \varepsilon^2)^{1/2} \frac{1}{(1 - \varepsilon \cos E)} dE \\
 &= a(1 - \varepsilon^2)^{1/2} \frac{1}{r} dE \quad (B.31)
 \end{aligned}$$

Then, the differential area becomes

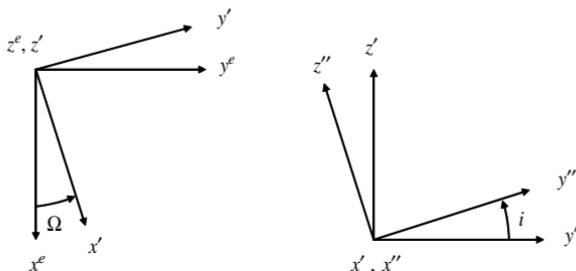
$$dA = \frac{1}{2} r^2 df = \frac{1}{2} a(1 - \varepsilon^2)^{1/2} r dE = \frac{1}{2} a^2 (1 - \varepsilon^2)^{1/2} (1 - \varepsilon \cos E) dE \quad (B.32)$$

which, after integrating, yields

$$A = \frac{1}{2} a^2 (1 - \varepsilon^2)^{1/2} (E - \varepsilon \sin E) \quad (B.33)$$

This is the same result as obtained from the preceding geometrical approach.

The preceding equations provide the location within the orbital plane. This position must be transformed into the Earth-centered frame. This is accomplished by defining the following sequence of transformations. The rotation from the Earth-centered frame to an intermediate axis through the longitude of the ascending node  $\Omega$  and the rotation from this intermediate frame to the orbital plane through the orbit inclination angle  $i$  follows from Figs. B.1.



**Fig. B.1 Rotations from Earth-centered frame to orbital plane.**

## B.2 Orbital Velocity in Earth-Centered Frame

The orbit radius vector can be written in components defined by the unit vector defined in Fig. 7.5 as

$$\mathbf{r} = r \mathbf{e}_r = r \cos \theta \mathbf{e}_x + r \sin \theta \mathbf{e}_y \quad (\text{B.34})$$

where  $x$  and  $y$  are nonrotating inertial axes. The velocity in this nonrotating frame is just the derivative of Eq. (B.34). This derivative becomes

$$\mathbf{v} = \dot{\mathbf{r}} = (\dot{r} \cos \theta - r \sin \theta \dot{\theta}) \mathbf{e}_x + (\dot{r} \sin \theta + r \cos \theta \dot{\theta}) \mathbf{e}_y \quad (\text{B.35})$$

From Eq. (B.13), the radius is expressed as

$$r = \frac{h^2/\mu}{1 + \varepsilon \cos f} \quad (\text{B.36})$$

The derivative of this expression is obtained following the sequence shown next. In this sequence, from Eq. (B.1), the angle rate  $\dot{\theta}$  is related to the angular momentum by

$$\dot{\theta} = \frac{h}{r^2} \quad (\text{B.37})$$

and the true anomaly rate is equivalent to this angular rate:

$$\dot{f} \equiv \dot{\theta} \quad (\text{B.38})$$

With these relationships, the radius derivative becomes

$$\begin{aligned} \dot{r} &= \frac{(1 + \varepsilon \cos f) \frac{d}{dt}(h^2/\mu) - (h^2/\mu)(-\varepsilon \sin f \dot{f})}{(1 + \varepsilon \cos f)^2} \\ &= \frac{(h^2/\mu)(\varepsilon \sin f \dot{f})}{(1 + \varepsilon \cos f)(1 + \varepsilon \cos f)} \\ &= r \frac{(\varepsilon \sin f)}{(1 + \varepsilon \cos f)} \frac{h h/\mu}{r^2 h/\mu} \\ &= \frac{\mu}{h} (\varepsilon \sin f) \frac{(h^2/\mu)}{(1 + \varepsilon \cos f)} \frac{1}{r} \\ &= \frac{\mu}{h} \varepsilon \sin f \end{aligned} \quad (\text{B.39})$$

and the radius/angle rate product is found to be

$$r\dot{\theta} = \frac{h}{r} = \frac{\mu}{h}(1 + \varepsilon \cos f) \quad (\text{B.40})$$

Substituting these expressions into Eq. (B.35), the velocity becomes

$$\begin{aligned} v &= \left[ \frac{\mu}{h} \varepsilon \sin f \cos \theta - \sin \theta \frac{\mu}{h} (1 + \varepsilon \cos f) \right] e_x \\ &\quad + \left[ \frac{\mu}{h} \varepsilon \sin f \sin \theta + \cos \theta \frac{\mu}{h} (1 + \varepsilon \cos f) \right] e_y \\ &= \left[ \frac{\mu}{h} \varepsilon (\sin f \cos \theta - \cos f \sin \theta) - \frac{\mu}{h} \sin \theta \right] e_x \\ &\quad + \left[ \frac{\mu}{h} \varepsilon (\sin f \sin \theta + \cos f \cos \theta) + \frac{\mu}{h} \cos \theta \right] e_y \\ &= \left[ -\frac{\mu}{h} \varepsilon \sin(f - \theta) - \frac{\mu}{h} \sin \theta \right] e_x + \left[ \frac{\mu}{h} \varepsilon \cos(f - \theta) + \frac{\mu}{h} \cos \theta \right] e_y \\ &= -\frac{\mu}{h} (\varepsilon \sin \omega + \sin \theta) e_x + \frac{\mu}{h} (\varepsilon \cos \omega + \cos \theta) e_y \\ &= -\frac{\mu}{h} (\sin \theta + \varepsilon \sin \omega) e_x + \frac{\mu}{h} (\cos \theta + \varepsilon \cos \omega) e_y \end{aligned} \quad (\text{B.41})$$

The transformation in Eq. (7.25) transforms velocity components into the Earth-centered frame as

$$\begin{aligned} \begin{bmatrix} v_x^e \\ v_y^e \\ v_z^e \end{bmatrix} &= \begin{bmatrix} c\Omega & -s\Omega ci & s\Omega si \\ s\Omega & c\Omega ci & -c\Omega si \\ 0 & si & ci \end{bmatrix} \begin{bmatrix} -\frac{\mu}{h}(s\theta + \varepsilon s\omega) \\ \frac{\mu}{h}(c\theta + \varepsilon c\omega) \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} -\frac{\mu}{h}[c\Omega(s\theta + \varepsilon s\omega) + s\Omega ci(c\theta + \varepsilon c\omega)] \\ -\frac{\mu}{h}[s\Omega(s\theta + \varepsilon s\omega) - c\Omega ci(c\theta + \varepsilon c\omega)] \\ \frac{\mu}{h} si(c\theta + \varepsilon c\omega) \end{bmatrix} \end{aligned} \quad (\text{B.42})$$



## Appendix C

### Coarse-Alignment Error Equations

The process of alignment is to establish the wander-azimuth angle. It will be assumed that the error associated with the  $C_b^n$  azimuth rotation, the  $z$ -tilt error, is zero:  $\phi_z \equiv 0$ , that is, all azimuth error is attributed to the wander-azimuth angle.

#### C.1 Position Error Equations

Equation (13.27) is expanded to represent errors in north and east position, angular position errors, and errors in  $\sin \alpha$  and  $\cos \alpha$  terms

$$\begin{aligned}
C_e^n &= \bar{C}_g^n \bar{C}_e^g \\
&= \left\{ \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\} \begin{bmatrix} 1 & 0 & -\delta\theta_e \\ 0 & 1 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 1 \end{bmatrix} C_e^g \\
&\approx C_g^n \begin{bmatrix} 1 & 0 & -\delta\theta_e \\ 0 & 1 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 1 \end{bmatrix} C_n^g C_e^n + \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} C_n^g C_e^n \\
&= C_e^n + C_g^n \begin{bmatrix} 0 & 0 & -\delta\theta_e \\ 0 & 0 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 0 \end{bmatrix} C_n^g C_e^n + \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} C_n^g C_e^n \\
&= C_e^n + \begin{bmatrix} 0 & 0 & -\delta\theta_y \\ 0 & 0 & \delta\theta_x \\ \delta\theta_e & -\delta\theta_n & 0 \end{bmatrix} C_n^g C_e^n + \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} C_n^g C_e^n \\
&= C_e^n + \begin{bmatrix} \delta c\alpha & \delta s\alpha & -\delta\theta_y \\ -\delta s\alpha & \delta c\alpha & \delta\theta_x \\ \delta\theta_e & -\delta\theta_n & 0 \end{bmatrix} C_n^g C_e^n \\
&\equiv (I + EC_e^g) C_e^n
\end{aligned} \tag{C.1}$$

where the following have been used in between the third and fourth steps:

$$\delta\theta_x = \cos \alpha \delta\theta_n + \sin \alpha \delta\theta_e \quad (\text{C.2})$$

$$\delta\theta_y = -\sin \alpha \delta\theta_n + \cos \alpha \delta\theta_e \quad (\text{C.3})$$

## C.2 Position Error Dynamics

Position error dynamic equations, time-derivative expressions for angular position errors  $\delta\theta_x$  and  $\delta\theta_y$ , for coarse-alignment with  $\delta s\alpha$  and  $\delta c\alpha$  states are obtained in a similar manner as for the small perturbation angles. However, because the error representation is not in a convenient vector or skew-symmetric matrix form [see Eq. (13.29)], each component must be evaluated, rather than using generalized vector algebra.

First, consider the direction cosine matrix error

$$\begin{aligned} \delta C_e^n &= \bar{C}_e^n - C_e^n \\ &\equiv EC_e^g \end{aligned} \quad (\text{C.4})$$

The derivative of the second row of this equation is

$$\begin{aligned} \dot{\delta C}_e^n &= \dot{\bar{C}}_e^n - \dot{C}_e^n \\ &= \dot{E}C_e^g - E\Omega_{e/g}^g C_e^g \\ &= (\dot{E} - E\Omega_{e/g}^g)C_e^g \end{aligned} \quad (\text{C.5})$$

where

$$\dot{C}_e^g = -\Omega_{e/g}^g C_e^g \quad (\text{C.6})$$

and  $\Omega_{e/g}^g$  represents the skew-symmetric equivalent of the vector cross product  $(\omega_{e/g}^g \times)$ .

Returning to the first row in Eq. (C.4) and taking the derivative yield

$$\begin{aligned} \dot{\delta C}_e^n &= \bar{C}_e^n - \dot{C}_e^n \\ &= -\bar{\Omega}_{e/n}^n \bar{C}_e^n + \Omega_{e/n}^n C_e^n \\ &= -\bar{\Omega}_{e/n}^n \bar{C}_g^n \bar{C}_e^g + \Omega_{e/n}^n C_g^n C_e^g \\ &= -\bar{\Omega}_{e/n}^n \left\{ C_g^n + \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\} \begin{bmatrix} 1 & 0 & -\delta\theta_e \\ 0 & 1 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 1 \end{bmatrix} C_e^g + \Omega_{e/n}^n C_g^n C_e^g \\ &= \left( -\bar{\Omega}_{e/n}^n \left\{ C_g^n + \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\} \begin{bmatrix} 1 & 0 & -\delta\theta_e \\ 0 & 1 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 1 \end{bmatrix} + \Omega_{e/n}^n C_g^n \right) C_e^g \end{aligned} \quad (\text{C.7})$$

Equating (C.5) and (C.7),

$$\begin{aligned}
 \dot{E} - E\Omega_{e/g}^g &= -\bar{\Omega}_{e/n}^n \left\{ C_g^n + \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\} \begin{bmatrix} 1 & 0 & -\delta\theta_e \\ 0 & 1 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 1 \end{bmatrix} + \Omega_{e/n}^n C_g^n \\
 &\approx -\bar{\Omega}_{e/n}^n C_g^n - \bar{\Omega}_{e/n}^n \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
 &\quad - \bar{\Omega}_{e/n}^n C_g^n \begin{bmatrix} 1 & 0 & -\delta\theta_e \\ 0 & 1 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 1 \end{bmatrix} + \Omega_{e/n}^n C_g^n \\
 &\approx (\Omega_{e/n}^n - \bar{\Omega}_{e/n}^n) C_g^n \\
 &\quad - \Omega_{e/n}^n \left\{ \begin{bmatrix} \delta c\alpha & \delta s\alpha & 0 \\ -\delta s\alpha & \delta c\alpha & 0 \\ 0 & 0 & 0 \end{bmatrix} + C_g^n \begin{bmatrix} 1 & 0 & -\delta\theta_e \\ 0 & 1 & \delta\theta_n \\ \delta\theta_e & -\delta\theta_n & 1 \end{bmatrix} \right\} \\
 &= (\Omega_{e/n}^n - \bar{\Omega}_{e/n}^n) C_g^n - \Omega_{e/n}^n \begin{bmatrix} \delta c\alpha & \delta s\alpha & -\delta\theta_y \\ -\delta s\alpha & \delta c\alpha & \delta\theta_x \\ \delta\theta_e & -\delta\theta_n & 0 \end{bmatrix} \tag{C.8}
 \end{aligned}$$

Or

$$\dot{E} = E\Omega_{e/g}^g - \Omega_{e/n}^n E + (\Omega_{e/n}^n - \bar{\Omega}_{e/n}^n) C_g^n$$

The terms of interest are in the third column, first and second row, of this matrix. Expanding each term in this equation,

$$\begin{aligned}
 E\Omega_{e/g}^g &= \begin{bmatrix} \delta c\alpha & \delta s\alpha & -\delta\theta_y \\ -\delta s\alpha & \delta c\alpha & \delta\theta_x \\ \delta\theta_e & -\delta\theta_n & 0 \end{bmatrix} \begin{bmatrix} 0 & -\rho_d & \rho_e \\ \rho_d & 0 & -\rho_n \\ -\rho_e & \rho_n & 0 \end{bmatrix} \\
 &= \begin{bmatrix} \delta s\alpha\rho_d + \delta\theta_y\rho_e & -\delta c\alpha\rho_d - \delta\theta_y\rho_n & \delta c\alpha\rho_e - \delta s\alpha\rho_n \\ \delta c\alpha\rho_d - \delta\theta_x\rho_e & \delta s\alpha\rho_d + \delta\theta_x\rho_n & -\delta s\alpha\rho_e - \delta c\alpha\rho_n \\ -\delta\theta_n\rho_d & -\delta\theta_e\rho_d & \delta\theta_e\rho_e + \delta\theta_n\rho_n \end{bmatrix} \tag{C.9}
 \end{aligned}$$

$$\begin{aligned}
 \Omega_{e/n}^n E &= \begin{bmatrix} 0 & -\rho_z & \rho_y \\ \rho_z & 0 & -\rho_x \\ -\rho_y & \rho_x & 0 \end{bmatrix} \begin{bmatrix} \delta c\alpha & \delta s\alpha & -\delta\theta_y \\ -\delta s\alpha & \delta c\alpha & \delta\theta_x \\ \delta\theta_e & -\delta\theta_n & 0 \end{bmatrix} \\
 &= \begin{bmatrix} \rho_z\delta s\alpha + \rho_y\delta\theta_e & -\rho_z\delta c\alpha - \rho_y\delta\theta_n & -\rho_z\delta\theta_x \\ \rho_z\delta c\alpha - \rho_x\delta\theta_e & \rho_z\delta s\alpha + \rho_x\delta\theta_n & -\rho_z\delta\theta_y \\ -\rho_y\delta c\alpha - \rho_x\delta s\alpha & -\rho_y\delta s\alpha + \rho_x\delta c\alpha & \delta\theta_y\rho_y + \delta\theta_x\rho_x \end{bmatrix} \tag{C.10}
 \end{aligned}$$

and

$$\begin{aligned}
 (\Omega_{e/n}^n - \bar{\Omega}_{e/n}^n) C_g^n &= \left\{ \begin{bmatrix} 0 & -\rho_z & \rho_y \\ \rho_z & 0 & -\rho_x \\ -\rho_y & \rho_x & 0 \end{bmatrix} - \begin{bmatrix} 0 & -\bar{\rho}_z & \bar{\rho}_y \\ \bar{\rho}_z & 0 & -\bar{\rho}_x \\ -\bar{\rho}_y & \bar{\rho}_x & 0 \end{bmatrix} \right\} C_g^n \\
 &= \begin{bmatrix} 0 & \delta\rho_z & \rho_y - \bar{\rho}_y \\ -\delta\rho_z & 0 & -(\rho_x - \bar{\rho}_x) \\ -(\rho_y - \bar{\rho}_y) & \rho_x - \bar{\rho}_x & 0 \end{bmatrix} \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} -\delta\rho_z s\alpha & \delta\rho_z c\alpha & \rho_y - \bar{\rho}_y \\ -\delta\rho_z c\alpha & -\delta\rho_z s\alpha & -(\rho_x - \bar{\rho}_x) \\ -(\rho_e - \bar{\rho}_e) & \rho_n - \bar{\rho}_n & 0 \end{bmatrix} \quad (C.11)
 \end{aligned}$$

The wander-azimuth frame transport rate vector's Earth radii of curvature are approximated by a nominal value  $R$ . The velocity error in the wander-azimuth frame is represented in the perturbation form in Eq. (13.44). Substituting components of this equation into corresponding elements in transport rate equation yields for the transport rate errors

$$\rho_x - \bar{\rho}_x = -\frac{\delta v_y}{R} + \frac{v_y}{R^2} \delta h \quad (C.12)$$

$$\rho_y - \bar{\rho}_y = \frac{\delta v_x}{R} - \frac{v_x}{R^2} \delta h \quad (C.13)$$

where  $\delta R \approx \delta h$ .

Angular position dynamic error equations are obtained by equating elements in Eqs. (C.9–C.11) and using Eqs. (C.12) and (C.13). Equating the 1 : 3 elements in the  $E$  matrix equation, the following is obtained:

$$\delta \dot{\theta}_y = -\rho_e \delta c\alpha + \rho_n \delta s\alpha - \frac{\delta v_x}{R} + \frac{v_x}{R^2} \delta h \quad (C.14)$$

Equating the 2 : 3 elements, the following is obtained:

$$\delta \dot{\theta}_x = -\rho_e \delta s\alpha - \rho_n \delta c\alpha + \frac{\delta v_y}{R} - \frac{v_y}{R^2} \delta h \quad (C.15)$$

### C.3 Attitude Error Equations

The  $\phi$  form of the attitude error dynamics is (see Chapter 5)

$$\dot{\boldsymbol{\phi}} = \delta \boldsymbol{\rho} + \delta \boldsymbol{\omega}_{i/e}^n + \boldsymbol{\phi} \times \boldsymbol{\omega}_{i/n}^n + \boldsymbol{\epsilon}^n \quad (C.16)$$

Using the correspondence given in Eq. (13.31), the second term on the right side of this equation is expressed as

$$\begin{aligned}
 \delta\boldsymbol{\omega}_{i/e}^n &= EC_n^g \boldsymbol{\omega}_{i/e}^n = E\boldsymbol{\omega}_{i/e}^g \\
 &= \begin{bmatrix} \delta c\alpha & \delta s\alpha & -\delta\theta_y \\ -\delta s\alpha & \delta c\alpha & \delta\theta_x \\ \delta\theta_e & -\delta\theta_n & 0 \end{bmatrix} \begin{bmatrix} \Omega_n \\ 0 \\ \Omega_d \end{bmatrix} \\
 &= \begin{bmatrix} \delta c\alpha\Omega_n - \delta\theta_y\Omega_d \\ -\delta s\alpha\Omega_n + \delta\theta_x\Omega_d \\ \Omega_n s\alpha\delta\theta_x + \Omega_n c\alpha\delta\theta_y \end{bmatrix} \\
 &= \begin{bmatrix} \delta c\alpha\Omega_n - \delta\theta_y\Omega_z \\ -\delta s\alpha\Omega_n + \delta\theta_x\Omega_z \\ -\Omega_y\delta\theta_x + \Omega_x\delta\theta_y \end{bmatrix} \quad (C.17)
 \end{aligned}$$

Between the third and fourth lines  $\delta\theta_e$  was obtained from Eqs. (C.2) and (C.3), and the last line is obtained from the following equation:

$$\boldsymbol{\Omega}^n = C_g^n \boldsymbol{\Omega}^g = \begin{bmatrix} c\alpha & s\alpha & 0 \\ -s\alpha & c\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Omega_n \\ 0 \\ \Omega_d \end{bmatrix} = \begin{bmatrix} \Omega_n c\alpha \\ -\Omega_n s\alpha \\ \Omega_d \end{bmatrix} \equiv \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} \quad (C.18)$$

Assuming that the  $z$ -tilt error is zero,  $\phi_z = 0$ , the  $x$  and  $y$  components of the preceding tilt error equation become

$$\dot{\phi}_x = \frac{\delta v_y}{R} - \frac{v_y}{R^2} \delta h + \boldsymbol{\Omega}_n \delta c\alpha - \boldsymbol{\Omega}_z \delta\theta_y + \boldsymbol{\Omega}_z \phi_y + \varepsilon_x \quad (C.19)$$

and

$$\dot{\phi}_y = -\frac{\delta v_x}{R} + \frac{v_x}{R^2} \delta h - \boldsymbol{\Omega}_n \delta s\alpha + \boldsymbol{\Omega}_z \delta\theta_x - \boldsymbol{\Omega}_z \phi_x + \varepsilon_y \quad (C.20)$$

The  $\phi_z = 0$  assumption also implies

$$\dot{\phi}_x = 0 = \delta\rho_z - \boldsymbol{\Omega}_y \delta\theta_x + \boldsymbol{\Omega}_x \delta\theta_y + \omega_y \phi_x - \omega_x \phi_y + \varepsilon_z$$

or

$$\delta\rho_z = \boldsymbol{\Omega}_y \delta\theta_x - \boldsymbol{\Omega}_x \delta\theta_y - \omega_y \phi_x + \omega_x \phi_y - \varepsilon_z \quad (C.21)$$

#### C.4 Velocity Error Equations

The perturbation form of the velocity error equation is (see Chapter 5)

$$\begin{aligned}\delta\dot{\mathbf{v}}^n = \mathbf{v}^n \times (\delta\boldsymbol{\omega}_{e/n}^n + 2\delta\boldsymbol{\omega}_{i/e}^n) - (\boldsymbol{\rho} + 2\boldsymbol{\Omega}) \times \delta\mathbf{v}^n + \mathbf{f}^n \times \boldsymbol{\phi} + \delta\mathbf{f}^n \\ + \delta\mathbf{g}^n\end{aligned}\quad (\text{C.22})$$

Errors in transport rate and Earth rate were described earlier. Velocity error state components are presented in Sec. 13.2.2.

## Appendix D

### Fine-Alignment Error Equations

The coarse-alignment Kalman filter estimates errors in trigonometric functions  $\delta s\alpha$  and  $\delta c\alpha$ , allowing for large errors in the wander-azimuth angle. In this appendix, the large attitude error model is modified to represent small angles.

The wander angle error can be obtained by reexamining Appendix C's results. Equating the 1 : 1 elements in the  $E$  matrix equation results in the following:

$$\begin{aligned}
 \frac{d}{dt}(\delta c\alpha) &= \delta s\alpha\rho_d + \delta\theta_y\rho_e - \rho_z\delta s\alpha - \rho_y\delta\theta_e - \delta\rho_zs\alpha \\
 &= \delta s\alpha(\rho_d - \rho_z) + \rho_e\delta\theta_y - \rho_y\delta\theta_e - \delta\rho_zs\alpha \\
 &= -\dot{\alpha}\delta s\alpha + \rho_e\delta\theta_y - \rho_y\delta\theta_e - \delta\rho_zs\alpha
 \end{aligned} \tag{D.1}$$

Substituting from Eq. (C.21) and expressing the geographic-frame-referenced rate and position errors in the wander-azimuth frame yield the following:

$$\begin{aligned}
 \frac{d}{dt}(\delta c\alpha) &= -\dot{\alpha}\delta s\alpha + \rho_e\delta\theta_y - \rho_y\delta\theta_e - \delta\rho_zs\alpha \\
 &= -\dot{\alpha}\delta s\alpha + (s\alpha\rho_x + c\alpha\rho_y)\delta\theta_y - \rho_y(s\alpha\delta\theta_x + c\alpha\delta\theta_y) \\
 &\quad - (\Omega_y\delta\theta_x - \Omega_x\delta\theta_y - \omega_y\phi_x + \omega_x\phi_y - \varepsilon_z)s\alpha \\
 &= -\dot{\alpha}\delta s\alpha + s\alpha[(\rho_y + \Omega_y)\delta\theta_x - (\rho_x + \Omega_x)\delta\theta_y - \omega_y\phi_x + \omega_x\phi_y - \varepsilon_z] \\
 &= -\dot{\alpha}\delta s\alpha + s\alpha[\omega_y\delta\theta_x - \omega_x\delta\theta_y - \omega_y\phi_x + \omega_x\phi_y - \varepsilon_z]
 \end{aligned} \tag{D.2}$$

From Eq. (13.33), the left-hand-side of Eq. (D.2) is also equivalent to

$$\begin{aligned}
 \frac{d}{dt}(\delta c\alpha) &= -\left(\frac{d}{dt}s\alpha\right)\delta\alpha - s\alpha\left(\frac{d}{dt}\delta\alpha\right) \\
 &= -c\alpha\dot{\alpha}\delta\alpha - s\alpha\left(\frac{d}{dt}\delta\alpha\right) \\
 &= -\dot{\alpha}\delta s\alpha - s\alpha\left(\frac{d}{dt}\delta\alpha\right)
 \end{aligned} \tag{D.3}$$

Equating the corresponding terms in Eqs. (D.2) and (D.3), the following equation for wander-azimuth angle error is obtained:

$$\frac{d}{dt}(\delta\alpha) = \omega_y \delta\theta_x - \omega_x \delta\theta_y - \omega_y \phi_x + \omega_x \phi_y - \varepsilon_z \quad (\text{D.4})$$

The inertial navigation system error dynamic model for small error angles is completed using developments of Appendix C. Using Eqs. (13.32) and (13.33), the  $y$  component of position error from Eq. (C.14) becomes

$$\begin{aligned} \delta\dot{\theta}_y &= -\rho_e \delta c\alpha + \rho_n \delta s\alpha - \frac{\delta v_x}{R} + \frac{v_x}{R^2} \delta h \\ &= -\rho_e(-s\alpha\delta\alpha) + \rho_n(c\alpha\delta\alpha) - \frac{\delta v_x}{R} + \frac{v_x}{R^2} \delta h \\ &= (\rho_n c\alpha + \rho_e s\alpha)\delta\alpha - \frac{\delta v_x}{R} + \frac{v_x}{R^2} \delta h \\ &= \rho_x \delta\alpha - \frac{\delta v_x}{R} + \frac{v_x}{R^2} \delta h \end{aligned} \quad (\text{D.5})$$

The  $x$  component of position is obtained similarly.

Velocity and tilt error equations can be obtained using the following relationships:

$$\Omega_n \delta c\alpha = \Omega_y \delta\alpha \quad (\text{D.6})$$

$$\Omega_n \delta s\alpha = \Omega_x \delta\alpha \quad (\text{D.7})$$

Substituting these expressions into equations for the velocity and attitude errors in Appendix C yields the small-attitude error dynamic model presented in Sec. 13.2.2.

## Appendix E References

- [1] Britting, K. R., *Inertial Navigation Systems Analysis*, Wiley, New York, 1971.
- [2] Sage, A. P., *Optimal Systems Control*, Prentice-Hall, Englewood Cliffs, NJ, 1968.
- [3] Bierman, G. J., *Factorization Methods for Discrete Sequential Estimation*, Academic Press, Orlando, FL, 1977.
- [4] Department of Defense World Geodetic System, Defense Mapping Agency TR 8350.2, Washington, DC, Sept. 1987.
- [5] Conte, S. D., and de Boor, C., *Elementary Numerical Analysis: An Algorithmic Approach*, McGraw-Hill, New York, 1973, p. 33.
- [6] Bergeson, J. E., "Experiences with the B-1 Navigation Filter," *Practical Aspects of Kalman Filtering Implementation*, edited by G. T. Schmidt, AGARD Lecture Series No. 82, 1976, pp. 3-18.
- [7] Brumback, B. D., and Srinath, M. D., "A Fault-Tolerant Multisensor Navigation System Design," *IEEE Aerospace and Electronic Systems*, Vol. AES-23, No. 6, Nov. 1987, pp. 738-756.
- [8] Sommerfeld, A., *Optics*, Lectures on Theoretical Physics, Vol. IV, Academic Press, New York, 1950, pp. 80-81.
- [9] Åström, K. J., *Introduction to Stochastic Control Theory*, Academic Press, New York, 1970.
- [10] Prussing, J. E., and Conway, B. A., *Orbital Mechanics*, Oxford Univ. Press, New York, 1993, Chapter 9.
- [11] Julier, S., Uhlmann, J. K., and Durrant-Whyte, H. F., "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators," *IEEE Transactions on Automatic Control*, Vol. 45, No. 2, March 2000, pp. 477-482.
- [12] Julier, Simon J., and Uhlmann, Jeffrey K., "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, Vol. 92, No. 3, March 2004, pp. 401-422.
- [13] LaViola, J. J., "A Comparison of Unscented and Extended Kalman Filtering for Estimating Quaternion Motion," *Proceedings of the American Automatic Control Conference*, IEEE Press, Denver, CO, 2003, pp. 2435-2440.
- [14] El Sheimy, N., Shin, E., and Niu, X., "Kalman Filter Face-Off, Extended Versus Unscented Kalman Filters for Integrated GPS and MEMS-Based Inertial Systems," *Inside GNSS*, Vol. 1, No. 2, March 2006, pp. 48-54.

- [15] Norgaard, M., Poulsen, N. K., and Ravn, O., "New Developments in state Estimation for Nonlinear Systems," *Automatica*, Vol. 34, No. 11, 2000, pp. 1627–1638.
- [16] Wylie, C. R., Jr., *Advanced Engineering Mathematics*, 3rd ed., McGraw-Hill, New York, 1966, p. 97.
- [17] Kalman, R. E., "Fundamental Study of Adaptive Control Systems," *ASD-TR-61-27*, Vol. 1, U. S. Air Force Aeronautical Systems Division, Wright-Patterson AFB, OH, April 1962, pp. 61–63.
- [18] Anderson, B. D. O., and Moore, J. B., *Optimal Filtering*, Prentice-Hall, Upper Saddle River, NJ, 1979, p. 39.
- [19] Diesel, J. W., "Calibration of a Ring Laser Gyro Inertial Navigation System," Paper 87-7, *13th Biennial Guidance Test Symposium*, U. S. Air Force, Central Inertial Guidance Test Facility (CIGTF), Oct. 1987.
- [20] Soltz, J. A., Donna J. I., and Greenspan, R. L. "An Option for Mechanizing Integrated GPS/INS Solutions," *Journal of the Institute of Navigation*, Vol. 35, No. 4, June 1988, pp. 443–457.
- [21] Rogers, R. M., "IMU In-Motion Alignment without Benefit of Attitude Initialization," first published in Navigation, *Journal of the Institute of Navigation*, Vol. 44, No. 3, 1997, pp. 301–311.
- [22] Kelley, R. T., Kätz, I. N., and Bedoya, C. A., "Design, Development and Evaluation of an ADA Coded INS/GPS Open Loop Kalman Filter," *Inst. of Electrical and Electronics Engineers*, Vol. 1, May 1990, pp. 382–388.
- [23] Pham, T. M., "Kalman Filter Mechanization for INS Airstart," *Inst. of Electrical and Electronics Engineers, IEEE Aerospace and Electronic Systems Magazine*, Vol. 7, Issue 1, Jan. 1992, pp. 3–11.
- [24] Scherzinger, B. M., "Inertial Navigation Error Models for Large Heading Uncertainty," *Inst. of Electrical and Electronics Engineers, IEEE Position, Location and Navigation Symposium*, Atlanta, CA, Apr. 1996, pp. 477–484.
- [25] Martin, K. M., DeVries, T. W., MacDonald, D. A., and Merna, D. C., "Accommodating Large Azimuth Initialization Errors in an Integrated INS/GPS System," *24th Joint Services Data Exchange*, Nov. 1998.
- [26] Rogers, R. M., "Large Azimuth INS Error Models for In-Motion Alignment", *Institute of Navigation National Technical Meeting*, Long Beach, CA, Jan. 2001, pp. 172–184.
- [27] Bowditch, N., *American Practical Navigator*, Defense Mapping Agency, Pub. No. 9, Chapter VIII, 1984.
- [28] Lefferts, E. J., Markley, F. L., and Shuster, M. D., "Kalman Filtering for Spacecraft Attitude Estimation," *AIAA Jr. Guidance, Control, and Dynamics*, AIAA, New York, Vol. 5, No. 5, Sep.–Oct. 1982, pp. 417–429.
- [29] Pittelkau, M. E., "Kalman Filtering for Spacecraft System Alignment Calibration," *AIAA Jr. Guidance, Control, and Dynamics*, AIAA, Reston, VA, Vol. 24, No. 6, Nov.–Dec. 2001, pp. 1187–1195.
- [30] Gai, E., Daly, K., Harrison, J., and Lemos, L., "Star-Sensor-Based Satellite Attitude/Attitude Rate Estimator," *AIAA Jr. Guidance, Control, and Dynamics*, AIAA, Washington, DC, Vol. 8, No. 5, Sep.–Oct. 1985, pp. 560–565.
- [31] Psiake, M. L., Martel, F., and Parimal, K. P., "Three-Axis Attitude Determination via Kalman Filtering of Magnetometer Data," *AIAA Jr. Guidance,*

*Control, and Dynamics*, AIAA, Washington, DC, Vol. 13, No. 3, May–June 1990, pp. 506–514.

[32] Huddle, J. R., “Inertial Navigation System Error Model Considerations in Kalman Filter Applications,” *Control and Dynamic Systems*, edited by C. T. Leondes, Vol. 20, Academic Press, New York, 1983, pp. 293–339.

[33] Rogers, R. M., “Velocity Error Representations in Inertial Navigation System Error Models,” *AIAA Guidance, Navigation and Control Conference*, AIAA Paper 95-3193, Aug. 1995, pp. 186–194.

[34] Pinson, J. C., “Inertial Guidance for Cruise Vehicles,” *Guidance and Control of Aerospace Vehicles*, edited by C. T. Leondes, McGraw-Hill, New York, 1963, pp. 113–187.

[35] Maybeck, P. S., *Stochastic Models, Estimation, and Control*, Vol. I, Academic Press, New York, 1979.

[36] Thompson, W. T., *Introduction to Space Dynamics*, Wiley, New York, 1961, p. 53.



## Appendix F Bibliography

### F.1 Inertial Sensors

Aronowitz, F., "The Laser Gyro," *Laser Applications*, Vol. 1, Academic Press, New York, 1971.

Craig, R. J. G., "Theory of Errors of a Multigimbal, Elastically Supported, Tuned Gyroscope," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-8, No. 3, May 1972, pp. 289–297.

Craig, R. J. G., "The Theory of Operation of an Elastically Supported, Tuned Gyroscope," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-8, No. 3, May 1972, pp. 280–288.

Ezekiel, S., and Arditty, H. J., "Fiber-Optic Rotation Sensors, Tutorial Review," *FiberOptic Rotation Sensors and Related Technologies*, Vol. 32, Springer-Verlag, Berlin, 1982.

Howe, E. W., and Savet, P. H., "The Dynamically Tuned Free Rotor Gyro," *Control Engineering*, June 1964, pp. 67–72.

Martin, G. J., "Gyroscopes May Cease Spinning," *IEEE Spectrum*, Vol. 23, Feb. 1986, pp. 48–53.

### F.2 Kalman Filtering

"Advances in the Techniques and Technology of the Application on Nonlinear Filters and Kalman Filters," NATO AGARDograph 256, London, Mar. 1982.

Anderson, B. D. O., and Moore, J. B., *Optimal Filtering*, Prentice-Hall, Upper Saddle River, NJ, 1979.

Kalman, R. E., "Fundamental Study of Adaptive Control Systems," ASD TR-61-27, Wright-Patterson AFB, OH, Apr. 1962.

"Practical Aspects of Kalman Filtering Implementation," NATO AGARD Lecture Series 82, London, Mar. 1976.

Sage, A. P., and Melsa, J. L., *Estimation Theory with Applications to Communications and Control*, McGraw-Hill, New York, 1971.

Stengel, R. F., *Stochastic Optimal Control*, Wiley, New York, 1986.

"Theory and Applications of Kalman Filtering," NATO AGARDograph 139, London, Feb. 1970.

### F.3 Navigation Systems

“Advances in Strap-Down Inertial Systems,” NATO AGARD Lecture Series 133, London, Apr. 1984.

Britting, K. R., *Inertial Navigation Systems Analysis*, Wiley, New York, 1971.

Farrell, J. L., *Integrated Aircraft Navigation*, Academic Press, New York, 1976.

“Strap-Down Inertial Systems,” NATO AGARD Lecture Series 95, London, May 1978.

### F.4 Orbital Mechanics

Battin, R. H., *An Introduction to the Mathematics and Methods of Astrodynamics*, AIAA Education Series, AIAA, New York, 1987.

# Index

- Aerodynamic drag variations, inertial position, 264
- AHRS. *See* attitude heading reference system
- Aircraft simulation, 342
- Algebra
  - matrix, 7–47
  - vector, 7–47
- Alignment filter corrections
  - attitude, 321
  - position, 320–321
  - velocity, 321
- Alignment system error equations,
  - system error dynamics, 315–319
  - large-azimuth error model, 315–319
- Alternative velocity error, 114–116
  - attitude, 115
  - equations, 116
  - position, 115–116
- Angle errors, 62–64
- Angular position, 62–64
  - latitude angle errors, 62–64
  - longitude angle errors, 62–64
  - wander angle errors, 62–64
- Angular velocity addition theorem, 30–32
- Astronomical triangle, 25–26
- Attitude, 308, 309, 311
- Attitude change and observation, 341–342, 343
  - attitude iteration algorithms, 343
- Attitude corrections, 321
  - algorithms, 345
- Attitude determination
  - attitude correction algorithms, 345
  - estimation and, 339–359
  - iteration, 344
    - algorithm comparisons, 345–347
    - terrestrial, 339–344
- Attitude equations, 105
- Attitude error, 115
  - equations, 107–110, 390
  - vector, quaternion vs., 350–351
- Attitude estimation, 347–359
  - Kalman filter error dynamic models, 347–350
  - alternate inertial attitude
    - error vector dynamics
    - equation development, 351–352
  - body-referenced attitude error
    - vector and dynamics, 350
  - direction cosine matrix, 349
  - inertial attitude error vector
    - dynamics, 349–350
  - quaternions, 348
    - attitude error vector
      - vs., 350–351
  - error
    - dynamics, 348–349
    - vector forms, 355, 357
  - simulation comparisons, 357–359
- Attitude heading reference system (AHRS), 281
- Attitude initialization, benefit of, 315–324
- Attitude iteration algorithms, 343
- Azimuth
  - frame, wander, 52–53
  - variation, inertial position, 266
  - wander, 56
- Baro-inertial
  - errors, 118
  - feedback coefficients, 118–120
  - stability, 119–120

- Benefit of attitude initialization,  
315–324  
alignment  
    filter corrections, 320–321  
    system error equations, 315–319  
navigation equation initialization, 315  
observations, 319
- Body frame, 54  
Body to navigation frame, 58–59  
Body-referenced quaternion error vector,  
    353–354  
    dynamics, 350, 354  
    observations, 354–355
- Coarse-alignment error equations,  
    387–392  
    attitude, 390–391  
    position, 387–388  
        dynamics, 388–390  
        velocity, 392  
Calculus, 13–17  
Combined estimates of Kalman filtering,  
    204–207  
Continuous time, 141–145  
    discrete-time, 143  
Coordinate frame transformations,  
    54–64  
    body to navigation frame, 58–59  
    earth-centered, earth-fixed to  
        local geodetic, 56  
        local geodetic, 57  
        navigation frame, 56  
    earth-centered, inertial to local  
        geodetic frame, 57–58  
    geodetic to line of sight  
        transformation  
        matrix, 59–64  
Coordinate systems, 49–54  
    angular position, 62–64  
    body frame, 54  
    earth to navigation frame direction  
        cosine matrix, 61–62  
    earth-centered, earth-fixed (ECEF)  
        frame, 49–50  
    earth-centered, inertial frame, 50  
    geographic local horizon line of  
        sight, 54  
    local geodetic frame, 50  
    transformations and, 49–69  
    wander azimuth frame, 52–53  
Coordinate transformations, 19–28  
    right-handed sense, 22
- Data averaging, 203–204  
Data-collection rotation sequences,  
    239–240  
DCM. *See* direction cosine matrices  
Dead reckoning navigation / differential  
    global positioning system  
        (DR/DGPS), 327–328  
    implementation of, 332–333  
    testing, 333–337  
        nominal, 333–334  
        disrupted, 334–337  
Dead reckoning navigation equations,  
    328–329  
Dead reckoning system error model,  
    329–332  
    heading error, 331  
    navigation filter error dynamics  
        summary, 332  
    velocity error, 329–330  
Derivative-free estimation, 207–221  
    divided difference filter, 216–221  
    unscented Kalman filter, 207–215  
Differential global positioning system  
    position observations, 332  
Direction cosines, 26–28  
Direction cosine matrices  
    (DCM), 19–30  
    astronomical  
        positions, 24–28  
        triangle, 25–26  
    coordinate transformations, 19–28  
    elevation angles, 24–28  
    local level azimuth, 24–28  
    other dynamical descriptions, 30–32  
    rate of change, 28–30  
Direction cosine matrix, 349  
    determination, 340–341  
    properties, 21  
Discrete linear minimum variance  
    estimator, 191–194  
    Joseph's form, 194  
Discrete-time, 143  
Disrupted differential global positioning  
    system testing, 334–337  
Divided difference filter, 216–221  
    second-order, 218  
Doppler divergence, 156  
Doppler velocity sensors (DVS),  
    154–157  
    functionality, 154–155  
    outputs, navigation filter implemen-  
        tation, 155–156

- perturbation velocity error representation, 156–157
- DR/DGPS. *See* dead reckoning navigation/ differential global positioning system
- DVS. *See* Doppler velocity sensors
- Dynamic model extensions, 344
- Dynamic systems, 187–191
- Earth centered, earth fixed (ECEF) frame, 49–50  
local geodetic, 57  
navigation frame, 56  
solution, 291
- Earth centered earth-fixed to local geodetic, 56
- Earth centered frame  
orbital dynamics and, 377–383  
Kepler's equation, 381  
orbital velocity and, 384–385
- Earth centered inertial (ECI),  
frame, 50  
local geodetic frame, 57–58  
X-Y-Z coordinates, 72–75
- Earth models, 71–99  
axial symmetry, 84  
ellipsoid geometry, 71–79  
ellipsoid gravity, 79–87
- Earth relative velocity, 103
- Earth to navigation frame direction cosine matrix, 61–62  
wander angle rate, 61–62
- ECEF. *See* earth-centered, earth fixed
- ECI. *See* earth centered inertial
- Elevation angle, 24–28, 165–166
- Elevation variation, inertial position, 267
- Ellipsoid geometry  
earth centered X-Y-Z coordinates, 72–75  
radii of curvature, 75–79  
reference constants, 76, 77
- Ellipsoid gravity, 79–87  
gravitation potential, 79–84  
gravitational acceleration, 84–86  
gravity, 86–87
- Error covariance matrix, 184
- Error equations, 116
- Expectation, 16
- Exponential matrix, 19
- Feedback coefficients, 118–120
- Fiber-optic gyro (FOG), 139–140
- Filter estimate applications, 311  
attitude, 311  
position, 311  
velocity, 311
- Fine-alignment error equations, 393–394
- Fine-alignment Kalman filter, 279–281
- Flight-test evaluations, 251–274  
optical tracking trajectory reconstruction, 252–256  
tactical-air-navigation/inertial-navigation-unit reconstruction, 256–260  
vehicle dynamics, 260–273
- FOG. *See* fiber-optic gyro
- Forward-looking infrared line of sight systems, 169–172  
linearizations, 171  
measurement formulation, 170  
system  
functionality, 169  
measurements, 170–172
- Frame inertial rates, 77–79  
geographic, 78–79  
latitude and longitude, 77  
local, 78
- GDOP. *See* geometric dilution of precision
- Geodetic frame, local, 57–58
- Geodetic line of sight transformation matrix, 59–64  
relative positions, 60–61
- Geodetic, local, 56, 57
- Geographic frame inertial rate, 78–79
- Geographic local horizon line of sight (LOS), 54
- Geometric dilution of precision (GDOP), 166–169
- Global-positioning-system (GPS), 4, 160–169  
functionality, 160–165  
latitude and longitude positions measurements, 374–375
- receiver, 289–304  
inertial-navigation-system configuration  
Kalman filter, 290–299  
tuning parameters, 294  
Kalman filter configurations, 299–304  
filter, 290

- Global-positioning-system (GPS)  
*(Continued)*  
 satellite selection, 165–169  
   elevation angle, 165–166  
   geometric dilution of precision (GDOP), 166–169  
   horizontal dilution of precision (HDOP), 166–169  
 GPS. *See* global positioning system  
 Gravitation potential, 79–84  
 Gravitational acceleration, 84–86  
 Gravity, 86–87  
   anomalies of, 87  
   WGS model, 86  
 Gyro performance characterization, 135–140  
   fiber-optic gyro (FOG), 139–140  
   ring laser gyro (RLG), 135–139
- HDOP. *See* horizontal dilution of precision
- Heading error, 331
- Higher-order gravitation variation, 263–264
- Horizontal dilution of precision (HDOP), 166–169
- IMU. Inertial measurement unit
- Inertial attitude error vector dynamics, 349–350  
   equation development, 351–352  
   observations, 352  
   perturbation quaternion vectors, 353–355
- Inertial measurement unit (IMU), 3
- Inertial navigation system (INS), 4
- Inertial navigation system configuration, Kalman filter, 290–299  
   dynamic model summary, 299  
   earth-centered, earth-fixed navigation solution, 291  
   inertial-navigation-system filter error dynamic model, 291–299  
   tuning parameters, 294
- Inertial navigation system filter error dynamic model, 291–299
- Inertial navigation system ground alignment, 275–287  
   fine-alignment Kalman filter, 279–281  
   initial coarse alignment, 276–279  
   simulated ground fine, 281–287
- Inertial navigation system Kalman filter operation, 299–304
- Inertial navigation unit, attitude heading reference system (AHRS), 281
- Inertial position, 264  
   azimuth variation and, 266  
   elevation variation and, 267
- Inertial referenced quaternion error vector dynamics, 354
- Inertial sensors, 237–250
- Inertial velocity, 264
- Initial coarse alignment, 276–279  
   stationary alignment, 276–277
- In-motion alignment, 305–325  
   results, 321–324  
   transfer, 306–315  
   without benefit of attitude initialization, 315–324
- INS. *See* inertial navigation system
- Instability, 120
- Integrated differential global positioning system, dead reckoning navigation, 327–338
- Iteration algorithm comparisons, 345–347
- Iteration, attitude determination by, 344
- Janus configuration, 155
- Joseph's form, standard Kalman form, 194
- Kalman filter, 181–234  
   combined estimates, 204–207  
   configurations, 290  
     inertial-navigation-system configuration filter, 290  
   derivative-free estimation, 207–221  
   discrete linear minimum variance estimator, 191–194  
   equations  
     linear discrete, 184  
       *U-D* factored form, 194–201  
     fine-alignment, 279–281  
     inertial-navigation-system configuration, 290–299  
     recursive weighted least square (RWLS), 182–197  
       dynamic systems, 187–191  
     summed measurements, 201–204  
     tactical-air-navigation/inertial-navigation-unit, 256–258  
     unscented, 207–215

- Large-azimuth error model, 315–319  
Latitude angle errors, 62–64  
Latitude frame inertial rates, 77  
Least-squares parameter estimation, 14–17  
Legendre polynomial, 81  
Lemma, matrix inversion, 12–13  
Lightweight optical tracking system (LOTS), 252–256  
Line of sight (LOS), 54  
Line of sight angles transformation matrix, relative positions, 60–61  
Line of sight systems, forward-looking infrared, 169–172  
Line of sight transformation matrix, geodetic to, 59–64  
Linear discrete Kalman filter equations, 184  
Linearization, 60–61 techniques, 17–19  
Local frame inertial rates, 78  
Local geodetic, 56–57 frame, 50, 57–58  
Local level azimuth, 24–28  
Local-level navigation-frame equations, 103–105 attitude, 105 position equations, 105 velocity, 103–105  
Longitude angle errors, 62–64  
Longitude frame inertial rates, 77  
Longitude of the ascending node rate, 164  
LOS. *See* line of sight  
LOTS. *See* lightweight optical tracking system
- Matrices, 8 direction cosine (DCM), 19–30  
Matrix algebra, 7–47  
Matrix inversion lemma, 12–13  
Matrix/vector product, 10  
Monte Carlo simulation, 147, 148
- Navigation aids, 153–180 Doppler velocity sensors (DVS), 154–157 forward-looking infrared line of sight systems, 169–172 global-positioning-system (GPS) range, 160–169 tactical-air-navigation (TACAN) range, 157–160
- Navigation equations, 306–308 attitude, 308 filter estimate applications, 311 initialization, 315 position, 306–307 velocity, 307–308  
Navigation filter error dynamics summary, 332  
Navigation filter implementation, 155–158  
Navigation frame (wander azimuth), 56 equations, 103–105  
Navigation mechanization review, 238  
Navigation sensor models, 135–152 error, 140–149 gyro performance characterization, 135–140  
Navigation system error equations, 106–113, 308–311 alternative velocity, 114–116 attitude, 107–110, 309 formulation, 113 model, 365 position, 365–369  $\Psi$  attitude, 369–370 solution corrections, 373–374 system error dynamic model, 370–373 perturbation, 111 position, 110–111, 308–309 Schuler period, 111–113 velocity, 106–107, 309  
Numerical integration, 204
- Observation equations, 241–244 errors, 245 solutions for, 246 rotation 1, 241 rotation 2, 242 rotation 3, 243–244  
Optical tracking trajectory reconstruction, 252–256 lightweight, 252–256  
Orbital dynamics, 377–385 earth-centered frame, 377–383 Kepler's equation, 381 velocity, 384  
Orbital parameters, 164
- Perturbation errors, 111  
Perturbation quaternion vectors, 353–355 body-referenced quaternion error dynamics, 354 vector, 353–354

- Perturbation quaternion vectors  
*(Continued)*  
 inertial-referenced quaternion error  
 vector dynamics, 354
- Perturbation velocity error  
 representation, 156–157
- Pinson error model, 365–375  
 global positioning system latitude and  
 longitude, 374–375  
 navigation system error model,  
 365–369  
 navigation solution corrections,  
 373  
 position error, 365–369  
 $\psi$  attitude error, 369–370  
 system error dynamic model  
 summary, 370–373
- Position, 306–309, 311  
 equations, 105  
 error, 115–116  
 dynamics, 388–390  
 equations, 110–111, 387–388
- Position/velocity equations, 261–262
- Position/velocity error equations,  
 262–264  
 aerodynamic drag variation, 264  
 higher-order gravitation variation,  
 263–264  
 right-hand side, 262
- Processing sequences, 244, 247
- $\Psi$  attitude, 369–370
- $\Psi$  formulation, 113
- Quaternion, 348  
 attitude error vector vs., 350–351  
 corrections, 355–357  
 error dynamics, 348–349  
 error vector forms, 355, 357  
 corrections, 355, 357
- Radar position observation, 265–266  
 linearizations, 266–267  
 azimuth variation, 266  
 elevation variation, 267
- Radar tracking trajectory reconstruction,  
 260–273  
 measurements, 267  
 position/velocity  
 equations, 261–262  
 error equations, 262–264  
 radar position observations, 265–266  
 linearizations, 266–267  
 sounding rocket trajectory, 268–273
- Radii of curvature, 75–79  
 frame inertial rates, 77–79
- Random constant errors, 140–141
- Random process dynamic, 141–149  
 continuous time, 141–145  
 stochastic differential equation  
 approach, 145–149
- Rate of change, direction cosine matrices  
 and, 28–30
- Recursive weighted least squares  
 (RWLS), 182–187  
 constant systems, 182–187  
 data trends, 186–187  
 weighted, 182
- dynamic systems, 187–191  
 error covariance matrix, 184  
 linear discrete Kalman filter  
 equations, 184
- Reference ellipsoid, 76, 77
- Relative positions,  
 linearization, 60–61, 170–172  
 line of sight angles transformation  
 matrix and, 60–61
- Right hand  
 sense, 22  
 side, 262
- Ring laser gyros (RLG),  
 135–139
- RLG. *See* ring laser gyros
- Rotation 1, 241, 242, 243–244
- RWLS. *See* recursive weighted lease  
 square.
- Sagnac experiment, 136
- Satellite selection, 165–169  
 elevation angle, 165–166  
 geometric dilution of precision  
 (GDOP), 166–169  
 horizontal dilution of precision  
 (HDOP), 166–169
- Scalar nonlinear system, example of,  
 212–215
- Schuler period, 111–113
- Second-order divided difference  
 filter, 218
- Sensor error model, 140–149, 238  
 data-collection rotation sequences,  
 239–240  
 random  
 constant, 140–141  
 process dynamic, 141–149  
 solutions for, 239
- Similarity transformation, 10, 30

- Simulated ground fine alignment, 281–287  
inertial navigation unit, 281
- Solution for errors, observation equations and, 246
- Sounding rocket trajectory reconstruction, 268–273
- Standard kalman form, 194
- State space vector, 8
- Stationary alignment, 276–277  
errors, 277–279
- Stirling interpolation formula, 216
- Stochastic differential equation approach, 145–149  
Monte Carlo simulation, 147, 148
- Strap-down inertial sensor laboratory calibration, 237–250  
observation equations, 241–244  
processing sequences, 244, 247  
sensor error model, 238  
simulated laboratory data calibration, 247–250
- Strap down navigation systems, 102
- Structural misalignments, 314–315
- Summed measurements, 201–204  
data averaging, 203–204  
numerical integration, 204
- Symmetric matrix, 11
- System error dynamics, 315–319  
large-azimuth error model, 315–319
- TACAN. *See* tactical-air-navigation
- Tactical-air-navigation (TACAN)  
filter outputs, 258  
functionality of, 157  
measurements, 258  
navigation filter implementation, 157–158  
range, 157–160  
measurement mix, 158–160
- Tactical-air-navigation/inertial-navigation-unit Kalman filter states, 256–258  
flight test results, 258–259
- Taylor series expansion, 17
- Terrestrial attitude-determination, 339–344  
aircraft simulation, 342  
attitude change and observation, 341–342  
direction cosine matrix determination, 340–341  
dynamic model extensions, 344
- Terrestrial navigation, 101–134  
error equations, 106–113  
local-level navigation-frame equations, 103–105  
strap-down systems, 102  
vertical channel, 116–120
- Three space vectors, 10–13  
matrix inversion lemma, 12–13  
symmetric matrix, 11
- Transfer alignment, 306–315  
navigation  
equations, 306–308  
filter estimate applications, 311  
system error equations, 308–311  
observations, 311  
process of, 306  
results, 312–313  
structural misalignments, 314–315  
test data overview, 312–313
- True covariance approximation, scalar example of, 210–212
- U-D* factored form  
measurement update, 195–196  
algorithm, 197–201  
time update, 196–197
- U-D* factorization, 11
- U-D* measurement update algorithm, 197–201
- Unscented Kalman filter, 207–215  
scalar nonlinear system example, 212–215  
true covariance approximation, 210–212
- Vector algebra, 7–47
- Vector, state-space, 8
- Vector/matrix algebra, three-space vectors, 10–13
- Vector/matrix calculus, 13–17  
least-squares parameter estimation, 14–17
- Vector/matrix manipulation, 8  
matrix/vector product, 10  
similarity transformation, 10
- Vehicle dynamics, radar tracking trajectory reconstruction, 260–273
- Velocity, 307–308, 309, 311  
corrections, 321  
equations, 103–105, 392  
error, 329–330, 392

- Vertical channel, 116–120  
  feedback coefficients, 118–120  
  instability, 120
- Wander angle errors, 62–64
- Wander angle rate, 61–62
- Wander azimuth, 56  
  frame, 52–53
- Weighted least squares, 182  
  recursive, 182–187
- WGS gravity model, 86

## **Supporting Materials**

To download your software, and any software updates, please go to [www.aiaa.org/publications/supportmaterials](http://www.aiaa.org/publications/supportmaterials). Follow the instructions provided and enter the following password: **ellipsoid**.

Many of the topics introduced in this book are discussed in more detail in other AIAA publications. For a complete listing of titles in the AIAA Education Series, as well as other AIAA publications please visit <http://www.aiaa.org>.