

Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems

For a listing of recent titles in the *Artech House*
GNSS Technology and Applications Series, turn to the back of this book.

Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems

Paul D. Groves



**ARTECH
HOUSE**

BOSTON | LONDON
artechhouse.com

Library of Congress Cataloging-in-Publication Data

A catalog record for this book is available from the U.S. Library of Congress.

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library.

ISBN-13: 978-1-58053-255-6

Cover design by

© 2008 Paul D. Groves

All rights reserved.

All rights reserved. Printed and bound in the United States of America. No part of this book may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from the publisher.

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Artech House cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

10 9 8 7 6 5 4 3 2 1

Contents

Preface	xv
PART I	
Introduction	1
CHAPTER 1	
Introduction	3
1.1 What Is Navigation?	3
1.1.1 Position Fixing	4
1.1.2 Dead Reckoning	6
1.2 Inertial Navigation	7
1.3 Radio and Satellite Navigation	8
1.3.1 Terrestrial Radio Navigation	9
1.3.2 Satellite Navigation	10
1.4 Feature Matching	12
1.5 The Complete Navigation System	13
References	14
PART II	
Navigation Mathematics	15
CHAPTER 2	
Coordinate Frames, Kinematics, and the Earth	17
2.1 Coordinate Frames	17
2.1.1 Earth-Centered Inertial Frame	19
2.1.2 Earth-Centered Earth-Fixed Frame	20
2.1.3 Local Navigation Frame	20
2.1.4 Body Frame	21
2.1.5 Other Frames	22
2.2 Kinematics	23
2.2.1 Euler Attitude	24
2.2.2 Coordinate Transformation Matrix	26
2.2.3 Quaternion Attitude	29
2.2.4 Rotation Vector	30
2.2.5 Angular Rate	30

2.2.6	Cartesian Position	31
2.2.7	Velocity	33
2.2.8	Acceleration	34
2.3	Earth Surface and Gravity Models	35
2.3.1	The Ellipsoid Model of the Earth's Surface	36
2.3.2	Curvilinear Position	38
2.3.3	The Geoid and Orthometric Height	43
2.3.4	Earth Rotation	44
2.3.5	Specific Force, Gravitation, and Gravity	45
2.4	Frame Transformations	49
2.4.1	Inertial and Earth Frames	49
2.4.2	Earth and Local Navigation Frames	50
2.4.3	Inertial and Local Navigation Frames	51
2.4.4	Transposition of Navigation Solutions	52
	References	53
	Selected Bibliography	53
	Endnotes	54

CHAPTER 3

The Kalman Filter	55	
3.1	Introduction	55
3.1.1	Elements and Phases of the Kalman Filter	56
3.1.2	Kalman Filter Applications	58
3.2	Algorithms and Models	59
3.2.1	Definitions	59
3.2.2	Kalman Filter Algorithm	62
3.2.3	Kalman Filter Behavior	65
3.2.4	System Model	67
3.2.5	Measurement Model	70
3.2.6	Closed-Loop Kalman Filter	73
3.3	Implementation Issues	74
3.3.1	Tuning and Stability	74
3.3.2	Algorithm Design	75
3.3.3	Numerical Issues	77
3.3.4	Handling Data Lags	78
3.3.5	Kalman Filter Design Process	80
3.4	Extensions to the Kalman Filter	80
3.4.1	Extended and Linearized Kalman Filter	80
3.4.2	Time-Correlated Noise and the Schmidt-Kalman Filter	83
3.4.3	Adaptive Kalman Filter	85
3.4.4	Multiple-Hypothesis Filtering	86
3.4.5	Kalman Smoothing	90
	References	91
	Selected Bibliography	93
	Endnotes	93

PART III

Navigation Systems	95
--------------------	----

CHAPTER 4

Inertial Sensors	97
4.1 Accelerometers	98
4.1.1 Pendulous Accelerometers	100
4.1.2 Vibrating-Beam Accelerometers	101
4.2 Gyroscopes	101
4.2.1 Spinning-Mass Gyroscopes	102
4.2.2 Optical Gyroscopes	105
4.2.3 Vibratory Gyroscopes	108
4.3 Inertial Measurement Units	109
4.4 Error Characteristics	112
4.4.1 Biases	113
4.4.2 Scale Factor and Cross-Coupling Errors	114
4.4.3 Random Noise	115
4.4.4 Further Error Sources	117
4.4.5 Error Models	117
References	118

CHAPTER 5

Inertial Navigation	121
5.1 Inertial-Frame Navigation Equations	122
5.1.1 Attitude Update	123
5.1.2 Specific-Force Frame Transformation	124
5.1.3 Velocity Update	125
5.1.4 Position Update	126
5.2 Earth-Frame Navigation Equations	126
5.2.1 Attitude Update	126
5.2.2 Specific-Force Frame Transformation	128
5.2.3 Velocity Update	128
5.2.4 Position Update	129
5.3 Local-Navigation-Frame Navigation Equations	129
5.3.1 Attitude Update	130
5.3.2 Specific-Force Frame Transformation	132
5.3.3 Velocity Update	132
5.3.4 Position Update	133
5.3.5 Wander-Azimuth Implementation	134
5.4 Navigation Equations Precision	135
5.4.1 Iteration Rates	136
5.4.2 Attitude Update	137
5.4.3 Specific-Force Frame Transformation	142
5.4.4 Velocity and Position Updates	143
5.4.5 Effects of Vibration	144

5.5	Initialization and Alignment	146
5.5.1	Position and Velocity Initialization	146
5.5.2	Attitude Initialization	147
5.5.3	Fine Alignment	150
5.6	INS Error Propagation	151
5.6.1	Short-Term Straight-Line Error Propagation	152
5.6.2	Medium and Long-Term Error Propagation	154
5.6.3	Errors Due to Circling	157
5.7	Platform INS	157
5.8	Horizontal-Plane Inertial Navigation	158
	References	159
	Selected Bibliography	159
	Endnotes	160

CHAPTER 6

	Satellite Navigation Systems	161
6.1	Fundamentals of Satellite Navigation	161
6.1.1	GNSS Architecture	162
6.1.2	Positioning	163
6.1.3	Signals and Range Measurements	166
6.2	Global Positioning System	170
6.2.1	Space and Control Segments	171
6.2.2	Signals	173
6.2.3	Navigation Data Messages	176
6.2.4	Augmentation Systems	177
6.3	GLONASS	179
6.3.1	Space and Control Segments	179
6.3.2	Signals	180
6.3.3	Navigation Data Messages	181
6.4	Galileo	181
6.4.1	Space and Ground Segments	182
6.4.2	Signals	183
6.4.3	Navigation Data Messages	186
6.5	Regional Navigation Systems	186
6.5.1	Beidou and Compass	186
6.5.2	QZSS	187
6.5.3	IRNSS	188
6.6	GNSS Interoperability	188
6.6.1	Frequency Compatibility	189
6.6.2	User Competition	189
6.6.3	Multistandard User Equipment	190
	References	190
	Selected Bibliography	193

CHAPTER 7

	Satellite Navigation Processing, Errors, and Geometry	195
7.1	Satellite Navigation Geometry	196
7.1.1	Satellite Position and Velocity	196

7.1.2	Range, Range Rate, and Line of Sight	203
7.1.3	Elevation and Azimuth	207
7.1.4	Signal Geometry and Navigation Solution Accuracy	208
7.2	Receiver Hardware and Antenna	211
7.2.1	Antennas	212
7.2.2	Reference Oscillator and Receiver Clock	213
7.2.3	Receiver Front-End	214
7.2.4	Baseband Signal Processor	216
7.3	Ranging Processor	226
7.3.1	Acquisition	227
7.3.2	Code Tracking	229
7.3.3	Carrier Tracking	234
7.3.4	Tracking Lock Detection	240
7.3.5	Navigation-Message Demodulation	241
7.3.6	Carrier-Power-to-Noise-Density Measurement	242
7.3.7	Pseudo-Range, Pseudo-Range-Rate, and Carrier-Phase Measurements	244
7.4	Range Error Sources	245
7.4.1	Satellite Clock and Ephemeris Prediction Errors	245
7.4.2	Ionosphere and Troposphere Propagation Errors	247
7.4.3	Tracking Errors	250
7.4.4	Multipath	254
7.5	Navigation Processor	258
7.5.1	Single-Point Navigation Solution	259
7.5.2	Filtered Navigation Solution	262
7.5.3	Combined Navigation and Tracking	270
7.5.4	Position Error Budget	273
	References	274
	Selected Bibliography	277
	Endnotes	277

CHAPTER 8

	Advanced Satellite Navigation	279
8.1	Differential GNSS	279
8.1.1	Spatial and Temporal Correlation of GNSS Errors	279
8.1.2	Local and Regional Area DGNSS	280
8.1.3	Wide Area DGNSS	282
8.1.4	Precise Point Positioning	282
8.1.5	Relative GNSS	283
8.2	Carrier-Phase Positioning and Attitude	283
8.2.1	Integer Ambiguity Resolution	285
8.2.2	GNSS Attitude Determination	288
8.3	Poor Signal-to-Noise Environments	289
8.3.1	Antenna Systems	290
8.3.2	Receiver Front-End Filtering	291
8.3.3	Assisted GNSS	291

8.3.4	Acquisition	291
8.3.5	Tracking	293
8.3.6	Extended Coherent Integration	294
8.4	Multipath Mitigation	294
8.4.1	Antenna Systems	294
8.4.2	Receiver-Based Techniques	295
8.4.3	Multipath Mapping	296
8.4.4	Navigation Processor Filtering	296
8.5	Signal Monitoring	296
8.6	Semi-Codeless Tracking	297
	References	298

CHAPTER 9

	Terrestrial Radio Navigation	303
9.1	Point-Source Systems	303
9.2	Loran	305
9.2.1	The Loran Systems	306
9.2.2	Signals and User-Equipment Processing	307
9.2.3	Positioning	308
9.2.4	Error Sources	310
9.2.5	Differential Loran	311
9.3	Instrument Landing System	311
9.4	Urban and Indoor Positioning	312
9.4.1	Mobile Phones	312
9.4.2	Signals of Opportunity	313
9.4.3	GNSS Repeaters	314
9.4.4	WLAN Positioning	314
9.4.5	UWB Positioning	315
9.4.6	Short-Range Beacons	316
9.5	Relative Navigation	316
9.6	Tracking	318
9.7	Sonar Transponders	318
	References	318

CHAPTER 10

	Dead Reckoning, Attitude, and Height Measurement	321
10.1	Attitude Measurement	321
10.1.1	Leveling	321
10.1.2	Magnetic Heading	322
10.1.3	Integrated Heading Measurement	326
10.1.4	Attitude and Heading Reference System	327
10.2	Height and Depth Measurement	327
10.2.1	Barometric Altimeter	328
10.2.2	Depth Pressure Sensor	329
10.2.3	Radar Altimeter	329
10.3	Odometers	330

10.4 Pedestrian Dead Reckoning	335
10.5 Doppler Radar and Sonar	337
10.6 Other Dead-Reckoning Techniques	340
10.6.1 Image Processing	341
10.6.2 Landmark Tracking	341
10.6.3 Correlation Velocity Log	341
10.6.4 Air Data	342
10.6.5 Ship's Log	342
References	342
CHAPTER 11	
Feature Matching	345
11.1 Terrain-Referenced Navigation	345
11.1.1 Sequential Processing	346
11.1.2 Batch Processing	347
11.1.3 Performance	349
11.1.4 Laser TRN	350
11.1.5 Barometric TRN	351
11.1.6 Sonar TRN	351
11.2 Image Matching	351
11.2.1 Scene Matching by Area Correlation	352
11.2.2 Continuous Visual Navigation	353
11.3 Map Matching	353
11.4 Other Feature-Matching Techniques	355
11.4.1 Stellar Navigation	356
11.4.2 Gravity Gradiometry	356
11.4.3 Magnetic Field Variation	357
References	357
Selected Bibliography	359
PART IV	
Integrated Navigation	361
CHAPTER 12	
INS/GNSS Integration	363
12.1 Integration Architectures	364
12.1.1 Correction of the Inertial Navigation Solution	365
12.1.2 Loosely Coupled Integration	368
12.1.3 Tightly Coupled Integration	370
12.1.4 GNSS Aiding	371
12.1.5 Deep Integration	373
12.2 System Model and State Selection	375
12.2.1 State Selection and Observability	376
12.2.2 INS State Propagation in the Inertial Frame	378
12.2.3 INS State Propagation in the Earth Frame	382

12.2.4	INS State Propagation Resolved in the Local Navigation Frame	384
12.2.5	INS System Noise	387
12.2.6	GNSS State Propagation and System Noise	388
12.3	Measurement Models	389
12.3.1	Loosely Coupled Integration	390
12.3.2	Tightly Coupled Integration	393
12.3.3	Deep Integration	396
12.3.4	Estimation of Attitude and Instrument Errors	398
12.4	Advanced INS/GNSS Integration	399
12.4.1	Differential GNSS	399
12.4.2	Carrier-Phase Positioning and GNSS Attitude	399
12.4.3	Large Heading Errors	401
12.4.4	Advanced IMU Error Modeling	402
12.4.5	Smoothing	403
	References	403
	Selected Bibliography	406
	Endnotes	406

CHAPTER 13

INS Alignment and Zero Velocity Updates	407
13.1 Transfer Alignment	407
13.1.1 Conventional Measurement Matching	409
13.1.2 Rapid Transfer Alignment	410
13.1.3 Reference Navigation System	412
13.2 Quasi-Stationary Alignment with Unknown Heading	413
13.3 Quasi-Stationary Fine Alignment and Zero Velocity Updates	415
References	417
Selected Bibliography	418

CHAPTER 14

Multisensor Integrated Navigation	419
14.1 Integration Architectures	420
14.1.1 Least-Squares Integration	420
14.1.2 Cascaded Integration	422
14.1.3 Centralized Integration	424
14.1.4 Federated Integration	426
14.1.5 Hybrid Integration Architectures	429
14.1.6 Total-State Kalman Filter Employing Prediction	429
14.1.7 Error-State Kalman Filter	432
14.2 Terrestrial Radio Navigation	433
14.2.1 Loosely Coupled Integration	434
14.2.2 Tightly Coupled Integration	434
14.3 Dead Reckoning, Attitude, and Height Measurement	437
14.3.1 Attitude	438
14.3.2 Height and Depth	440

14.3.3 Odometers	441
14.3.4 Pedestrian Dead Reckoning	443
14.3.5 Doppler Radar and Sonar	444
14.4 Feature Matching	445
14.4.1 Position Fixes	445
14.4.2 Line Fixes	446
14.4.3 Ambiguous Measurements	448
References	448

CHAPTER 15

Fault Detection and Integrity Monitoring	451
15.1 Failure Modes	451
15.1.1 Inertial Navigation	452
15.1.2 GNSS	452
15.1.3 Terrestrial Radio Navigation	453
15.1.4 Dead Reckoning, Attitude, and Height Measurement	453
15.1.5 Feature Matching	453
15.1.6 Integration Algorithm	453
15.2 Range Checks	454
15.2.1 Sensor Outputs	454
15.2.2 Navigation Solution	455
15.2.3 Kalman Filter Estimates	455
15.3 Kalman Filter Measurement Innovations	455
15.3.1 Innovation Filtering	456
15.3.2 Innovation Sequence Monitoring	458
15.3.3 Remedyng Biased State Estimates	459
15.4 Direct Consistency Checks	460
15.4.1 Measurement Consistency Checks and RAIM	461
15.4.2 Parallel Solutions	463
15.5 Certified Integrity Monitoring	465
References	469

APPENDIX A

Vectors and Matrices	471
A.1 Introduction to Vectors	471
A.2 Introduction to Matrices	473
A.3 Special Matrix Types	476
A.4 Matrix Inversion	477
A.5 Calculus	478
References	478

APPENDIX B

Statistical Measures	479
B.1 Mean, Variance, and Standard Deviation	479
B.2 Probability Density Function	479
B.3 Gaussian Distribution	480

B.4 Chi-Square Distribution	481
References	483
List of Symbols	485
List of Acronyms and Abbreviations	497
About the Author	505
Index	507

Preface

This book has four main aims:

- To provide an introduction to navigation systems suitable for those with no prior knowledge;
- To describe the principles of operation of satellite, inertial, and many other navigation technologies, both qualitatively and mathematically;
- To review the state of the art in navigation technology;
- To provide a detailed treatment of integrated navigation.

It is aimed at professional scientists and engineers in industry, academia, and government, and at students at final year undergraduate, master's, and Ph.D. levels.

The book begins with a basic introduction to the main principles of navigation and a summary of the different technologies. The different coordinate frames, attitude representations, multiframe kinematics, Earth models, and gravity are then carefully explained, while the basic principles of each topic in the body of the book are explained before going into the details.

To cover the state of the art in navigation technology, the book goes beyond global navigation satellite systems (GNSS) and inertial navigation to describe terrestrial radio navigation, dead reckoning, and feature matching techniques. Topics covered include Loran, wireless local area network (WLAN) and ultra wideband (UWB) positioning, magnetometers, attitude and heading reference systems (AHRS), altimeters, odometers, pedestrian dead reckoning, Doppler radar and sonar, terrain-referenced navigation, image matching, and map matching.

The GNSS chapters describe the legacy and new Global Positioning System (GPS), Global Navigation Satellite System (GLONASS), and Galileo signals together and cover a range of advanced topics, including differential and carrier-phase positioning, GNSS attitude, multipath mitigation, and operation in poor signal-to-noise environments. Inertial navigation coverage includes accelerometer and gyroscope technology, navigation equations, initialization, alignment, and zero velocity updates.

Integrated navigation is served by a navigation-focused chapter on Kalman filtering, together with chapters on inertial navigation system (INS)/GNSS integration, including deep, and on multisensor integration. To support these, the chapters on the navigation sensors include comprehensive descriptions of the processing chains and error sources. The book concludes with a chapter on integrity monitoring, showing how a navigation system can detect and recover from faults.

The emphasis throughout is on providing an understanding of how navigation systems work, rather than on engineering details. The book focuses on the physical principles on which navigation systems are based, how they generate a navigation solution, how they may be combined, the origins of the error sources, and their mitigation. Later chapters build on material covered in earlier chapters, with comprehensive cross-referencing.

The book is divided into four parts. Part I comprises a nonmathematical introduction, while Part II provides the mathematical grounding to describe navigation systems and their integration. Part III describes the navigation systems, starting with inertial navigation, moving on to satellite navigation, and finishing with the other technologies. Part IV describes integrated navigation, including fault detection and integrity monitoring. Appendixes on vectors, matrices, and statistics, as well as full lists of symbols and acronyms complete the book.

Like many fields, navigation does not always adopt consistent notation and terminology. Here, a consistent notation has been adopted throughout the book, with common alternatives indicated where appropriate. The most commonly used conventions have generally been adopted, with some departures to avoid clashes and aid clarity.

Scalars are italicized and may be either upper or lower case. Vectors are lower-case bold, and matrices are uppercase bold, with the corresponding scalar used to indicate their individual components. The vector (or cross) product is denoted by \wedge , while Dirac notation (i.e., \dot{x} , \ddot{x} , and so on) is generally used to indicate time derivatives. All equations presented assume base SI units, the meter, second, and radian. Other units used in the text include the degree (1 degree = $\pi/180$ rad), the hour (1 hour = 3,600 seconds), and the g unit describing acceleration due to gravity ($1g \approx 9.8 \text{ ms}^{-2}$).

Unless stated otherwise, all uncertainties and error bounds quoted are ensemble 1σ standard deviations, which correspond to a 68 percent confidence level where a Gaussian (normal) distribution applies. This convention is adopted because integration and other estimation algorithms model the 1σ error bounds.

References are denoted by square brackets, numbered sequentially, and listed at the end of each chapter. Many chapters also include a selected bibliography, listing further publications of interest.

I would like to thank the following for their helpful comments and suggestions: Neil Boasman, Paul Cross, Steve Davison, Peter Duffett-Smith, Simon Gouldsworthy, Robin Handley, David Last, Nigel Mills, Washington Ochieng, Charles Offer, Tony Pratt, Graham Pulford, Andrew Runnalls, Andrey Soloviev, Roger Stokes, Jan Wendel, and Artech House's anonymous reviewer. I would like to thank QinetiQ for letting me reuse material I wrote for the *Principles of Integrated Navigation* course. This is marked by endnotes as QinetiQ copyright and appears mainly in Chapters 2, 3, 5, and 12. Finally, I would like to thank my family, friends, and colleagues for their patience and support.

A list of updates and corrections and printable symbol and acronym lists may be found online. See the links page at <http://www.artechhouse.com>.

PART I

Introduction

Introduction

What is meant by “navigation”? How do global navigation satellite systems (GNSS), such as the Global Positioning System (GPS), work? What is an inertial navigation system (INS)? This introductory chapter presents the basic concepts of navigation technology and provides a qualitative overview of the material covered in the body of the book. It introduces and compares the main navigation technologies and puts contemporary navigation techniques in a historical context.

Section 1.1 defines the concept of navigation and describes the two fundamental techniques that most navigation systems are based on: position fixing and dead reckoning. Section 1.2 provides a basic description of inertial navigation and discusses its pros and cons. Section 1.3 introduces radio navigation, covering both satellite and terrestrial systems. Section 1.4 discusses feature-matching techniques, such as terrain-referenced navigation (TRN). Finally, Section 1.5 discusses how different navigation technologies may be combined to produce the complete navigation system.

1.1 What Is Navigation?

There is no universally agreed definition of *navigation*. The *Concise Oxford Dictionary* [1] defines navigation as “any of several methods of determining or planning a ship’s or aircraft’s position and course by geometry, astronomy, radio signals, etc.” This encompasses two concepts. The first is the determination of the position and velocity of a moving body with respect to a known reference, sometimes known as the science of navigation. The second is the planning and maintenance of a course from one location to another, avoiding obstacles and collisions. This is sometimes known as the art of navigation and may also be known as guidance, pilotage, or routing, depending on the vehicle. This book is concerned only with the former concept, the science of navigation.

A *navigation technique* is a method for determining position and velocity, either manually or automatically. A *navigation system*, sometimes known as a navigation aid, is a device that determines position and velocity automatically. Some navigation systems also provide some or all of the attitude (including heading), acceleration, and angular rate. A navigation system may be self-contained aboard the navigating vehicle (e.g., an INS) or may require an external infrastructure as well as user components, such as radio navigation systems. The output of a navigation system or technique is known as the *navigation solution*. A *navigation sensor*

is a device used to measure a property from which the navigation system computes its navigation solution; examples include accelerometers, gyroscopes, and radio navigation receivers.

The navigation solution represents the coordinate frame of the navigating body (e.g., an aircraft, ship, car, or person) with respect to a reference coordinate frame. A common reference is the Earth. The components of the vectors comprising the navigation solution may be resolved about the axes of a third coordinate frame (e.g., north, east, and down). The coordinate frames commonly used for navigation are described in Section 2.1. Navigation systems often use a rotating reference frame to match the rotation of the Earth. This requires careful definition of the velocity and acceleration, which is covered in Section 2.2. That section also describes the different ways of representing attitude: as Euler angles, a coordinate transformation matrix, a quaternion, or a rotation vector. Section 2.3 describes how the surface of the Earth is modeled for navigation purposes and defines the latitude, longitude, and height.

Positioning is the determination of the position of a body, but not its velocity or attitude. Many navigation technologies, though strictly positioning systems, operate at a high enough rate for velocity to be derived from the rate of change of position.

Tracking or *surveillance* differs from navigation in that the position and velocity information is obtained by a third party without necessarily using equipment on board the object tracked. However, a tracking system may be used for navigation simply by transmitting the position and velocity measurements to the object that is being tracked. Similarly, a navigation system may be used for tracking by transmitting the navigation solution to a tracking station.

Most navigation techniques are based on either of two fundamental methods: position fixing and dead reckoning. These are described next.

1.1.1 Position Fixing

There are a number of position-fixing methods. Feature matching compares features at the current location, such as landmarks, waypoints, or terrain height, with a map to determine the current position. This is generally easier for a human than for a machine. Position fixes may also be obtained by measuring the ranges and/or bearing to known objects. This is illustrated for the two-dimensional case by Figure 1.1, whereby X marks the unknown user position, and A and B the known positions of two reference objects.

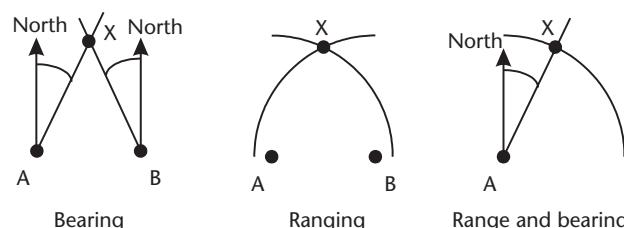


Figure 1.1 The bearing and ranging position fixing methods.

A two-dimensional position fix may be obtained by measuring the bearing to two known objects, where a bearing is the angle between the line of sight to an object and north (either true or magnetic). The user then lies along the intersection of the lines from the reference objects in the direction of each bearing measurement. The position fix may be extended to three dimensions by measuring the elevation angle to one of the reference objects, where the elevation is the angle between the line of sight to the object and a horizontal plane. For a given angular measurement accuracy, the accuracy of the position fix will degrade with distance from the reference objects.

If range measurements are taken from two known objects approximately in the same plane as the user, then the user position will lie on the intersection of two circles centered at the reference objects and with radii corresponding to the range measurements. However, there is generally a second intersection point. Often, prior information can be used to determine the correct position. Otherwise, a third range measurement is needed. If the range measurement accuracy is constant, there is no variation in position fix accuracy with distance from the reference objects. For a three-dimensional position fix, three range measurements are generally needed. There are still two points of intersection, but one is usually outside the operating range of the user. However, only a two-dimensional fix can be obtained when the reference objects and the user are within the same plane. This makes it difficult to obtain vertical position from a terrestrial ranging system. If both range and bearing measurements are made, a position fix can be obtained with a single reference object.

Bearing and elevation measurements can be made with relatively simple technology, such as a theodelite and magnetic compass. As well as terrestrial landmarks, the Sun, the Moon, and the stars can be used as reference objects. For example, the elevation of the Sun above the horizon at its highest point gives the latitude, whereas the timing of the sunrise or sunset with respect to the time at a known location can be used to determine longitude. Practical measurement of longitude on transoceanic voyages became possible in the 1760s, thanks to major advances in clock accuracy by John Harrison [2]. Bearing measurements can also be obtained from some radio navigation systems.

Ranging measurements can be made using radio signals, lasers, or radar. In passive ranging systems, the user receives signals from radio navigation transmitting stations, whereas in active ranging, the user transmits a signal to the reference object and receives either a reflection or a response transmission.

A component of a position fixing system external to the user's equipment is known as an *aid to navigation* (AtoN). AtoNs include landmarks, such as lights, and radio navigation signals.

Having obtained the range and/or bearing measurements, the position of the reference objects is required in order to determine the user position. Radio navigation transmitters will either broadcast their positions directly or be identifiable via a call sign or their frequency. Where the reference object is not a dedicated transmitter, it must be identified either manually or via an automated feature-matching technique, such as terrain-referenced navigation. These techniques are discussed in Section 1.4 and Chapter 11.

1.1.2 Dead Reckoning

Dead reckoning (possibly derived from “deduced reckoning”) either measures the change in position or measures the velocity and integrates it. This is added to the previous position in order to obtain the current position. The speed or distance traveled is measured in the body coordinate frame, so a separate attitude measurement is required to obtain the direction of travel in the reference frame. For two-dimensional navigation, a heading measurement is sufficient, whereas for three-dimensional navigation, a full three-component attitude measurement is needed. Figure 1.2 illustrates the concept of dead reckoning. Where the attitude is changing, the smaller the step size in the position calculation, the more accurate the navigation solution will be. The calculations were originally performed manually, severely limiting the data rate, but are now done by computer.

Traditional distance and velocity measurement methods include counting paces, using a pacing stick, and spooling a knotted rope off the back of the ship—hence the use of the knot as a unit of speed by the maritime community. Today, pace counting can be automated using a pedometer, while more sophisticated pedestrian dead reckoning (PDR) techniques using accelerometers also determine the step length. An odometer measures distance by counting the rotations of a wheel. Today, it is standard equipment on all road vehicles, but the technique dates back to Roman times. The equivalent for marine applications is a ship’s log comprising an underwater turbine, which rotates at a rate directly proportional to the ship’s speed. Contemporary velocity measurement methods include Doppler radar [3] and integrating accelerometer measurements within an inertial navigation system. Both technologies were established for military applications from the 1950s and civil applications from the 1960s, though an early form of inertial navigation was used on the German V2 rockets, developed during World War II. For marine applications, sonar may be used.

Height can be computed from pressure measurements using a barometric altimeter (baro). A radar altimeter (radalt) measures the height above the terrain, so can be used to determine an aircraft’s height where the terrain height is known.

Heading may be measured using a magnetic compass. This is an ancient technology, though today magnetic compasses and magnetometers are available with electronic readouts. Similarly, the Sun, Moon, and stars may be used to determine attitude where the time and approximate position are known. An INS obtains

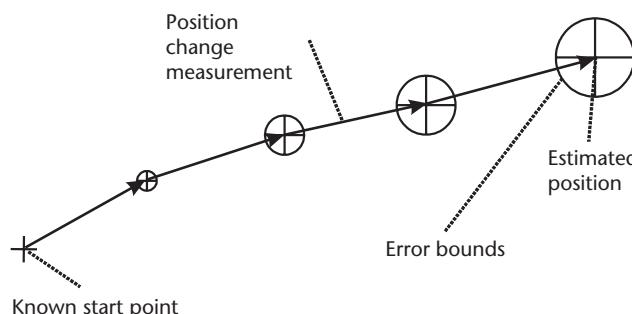


Figure 1.2 The dead reckoning method.

attitude by integrating angular rate measurements made by gyroscopes (gyros). The gyrocompass, a gyro-based heading sensor, was developed in the early part of the twentieth century. Inertial navigation is described in more detail in the next section and in Chapters 4 and 5, while the other dead reckoning techniques are covered in Chapter 10.

The dead reckoning position solution is the sum of a series of relative position measurements. Each of these will be subject to an error; consequently, the error in the position solution will grow with time. Position fixing does not suffer from error growth in the same way; it is reliant on components external to the user, which will generally not be continuously and universally available. Dead reckoning requires a known starting position, but after that will provide an uninterrupted navigation solution, bar equipment failure. Thus, the characteristics of dead reckoning and position fixing are complementary. A dead reckoning technique may be combined with one or more position fixing techniques in an *integrated navigation system* to get the benefits of both methods.

1.2 Inertial Navigation

An inertial navigation system (INS), sometimes known as an inertial navigation unit (INU), is a complete three-dimensional dead-reckoning navigation system. It comprises a set of inertial sensors, known as an *inertial measurement unit* (IMU), together with a navigation processor. The inertial sensors usually comprise three mutually orthogonal accelerometers and three gyroscopes aligned with the accelerometers. The navigation processor integrates the IMU outputs to give the position, velocity, and attitude. Figure 1.3 illustrates this.

The gyros measure angular rate, which is used by the navigation processor to maintain the INS's attitude solution. The accelerometers, however, measure specific force, which is the acceleration due to all forces except for gravity. Section 2.3.5 explains the concepts of gravity, gravitation, and specific force. In a strapdown INS, the accelerometers are aligned with the navigating body, so the attitude solution is used to transform the specific force measurement into the resolving coordinate frame used by the navigation processor. A gravity model is then used

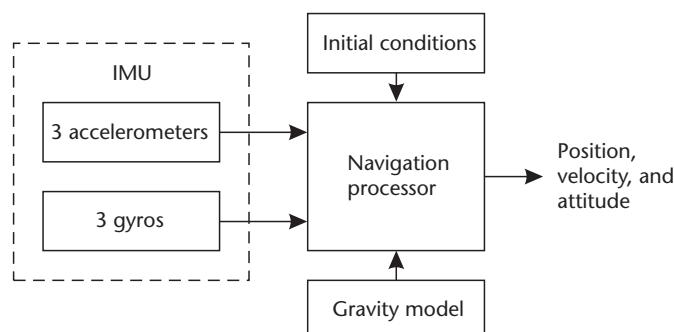


Figure 1.3 Basic schematic of an inertial navigation system. (From: [4]. © 2002 QinetiQ Ltd. Reprinted with permission.)

to obtain the acceleration from the specific force using the position solution. Integrating the acceleration produces the velocity solution, and integrating the velocity gives the position solution. The position and velocity for all INS and the heading for lower-grade systems must be initialized before the INS can compute a navigation solution. Inertial navigation processing is described in detail in Chapter 5.

Gyros and accelerometers are available in a range of different designs with significant variations in cost, size, mass, and performance. The higher-cost, higher-performance sensors are generally larger. Gyroscope designs fall into three main categories: spinning mass gyros, optical gyros, comprising ring laser gyros (RLGs) and interferometric fiber optic gyros (IFOGs), and vibratory gyros. Accelerometer technology comprises pendulous and vibrating beam accelerometers. Gyros and accelerometers manufactured using micro-electromechanical systems (MEMS) technology offer the advantages of low cost, size, and mass, and a high shock tolerance, but give relatively poor performance. Inertial sensors are described in more detail in Chapter 4.

The errors in an inertial navigation solution grow with time as successive accelerometer and gyro errors are summed. Feedback through the gravity model acts to stabilize the horizontal position and velocity, but destabilize the vertical channel. Overall navigation performance can vary by several orders of magnitude, depending on the quality of the inertial sensors. The highest grade is used mainly for ships, submarines, and some spacecraft. INSs used in military aircraft and commercial airliners exhibit a horizontal position error drift of less than 1,500m in the first hour and cost around \$100,000 or 100,000 Euros each. INSs used for light aircraft, helicopters, and guided weapons are typically two orders of magnitude poorer but cost much less. They are usually integrated with other navigation systems, such as GNSS. The cheapest and smallest MEMS inertial sensors are not suitable for inertial navigation at all, but may be used for PDR.

The principal advantages of inertial navigation are continuous operation, a high-bandwidth (at least 50 Hz) navigation solution, low short-term noise, and the provision of attitude, angular rate, and acceleration measurements, as well as position and velocity. The main drawbacks are the degradation in navigation accuracy with time and the cost.

1.3 Radio and Satellite Navigation

The first use of radio for navigation was in 1930 with the first navigation satellite launched in 1961. There are seven basic radio navigation techniques: marker beacons, direction finding, bearing/elevation, passive ranging, two-way ranging, hyperbolic ranging, and Doppler positioning. Marker beacons are the simplest technique—receiving a signal indicates that the user is in the vicinity of the transmitter.

With the direction finding, or angle of arrival (AOA), technique, the reference station is a simple beacon and may be used for other purposes, such as broadcasting. A rotatable directional antenna or phased array is then used by the receiver to obtain a bearing.

In the bearing or elevation technique, the reference station broadcasts a signal that varies with the direction of transmission, enabling the user to obtain a measurement of their bearing and/or elevation to the reference station without the need for a directional antenna. Examples include VHF omnidirectional radiorange (VOR) and the instrument landing system (ILS).

In a passive ranging, or time of arrival (TOA), system, such as GNSS, the reference station broadcasts a timing signal from which the user can deduce their range from the transmitter. This requires all of the clocks on both transmitters and receivers to be synchronized. However, the receiver clock can be synchronized simply by using an additional ranging signal to the number required to obtain a position solution.

In two-way ranging systems, such as distance measuring equipment (DME) and Beidou, the user transmits a call to the reference station, which then transmits back at a predetermined interval. This avoids the need for clock synchronization but introduces limitations to the number of users and repetition rate.

In a hyperbolic ranging, or time difference of arrival (TDOA), system, such as earlier versions of Loran (long-range navigation), the receiver measures the time difference (TD) in the signals broadcast by two transmitters, removing the need to synchronize the receiver clock. The locus of user position from a TD measurement is a hyperbola, giving the technique its name. Differential passive ranging, which uses a second receiver at a known location to calibrate transmitter synchronization and other common-mode errors, can also be described as a TDOA technique.

Doppler positioning relies on the transmitter moving along a prescribed trajectory. The receiver measures the Doppler shift of the received signal. When this is zero, the line-of-sight vector is perpendicular to the transmitter track, providing a position measurement in the direction parallel to the track. The perpendicular distance from the transmitter track is determined from the magnitude and rate of change of the Doppler shift.

1.3.1 Terrestrial Radio Navigation

The first radio navigation systems, used in the 1930s, were simple direction finders, used with broadcast stations, and 75-MHz marker beacons, used to delineate airways. Early bearing systems were also developed in this time. However, most terrestrial radio navigation technology had its origins in World War II [5]. The first hyperbolic systems, the American Loran-A and the British Decca and Gee systems, were developed around this time. The Decca navigation system closed in 2000, while Loran-A closed at the start of the 1980s and Gee closed in 1970. The longer range and more accurate Loran-C started in 1957. There were also B, D, and F variants of Loran, which are now obsolete. Loran-C coverage currently extends to the United States, most of Europe, and parts of Asia. With calibration, a positioning accuracy of a few tens of meters is obtained. Loran-C is now being upgraded to the new enhanced Loran (eLoran) standard. The Russian Chayka system is compatible with Loran and covers many parts of the former Soviet Union.

Omega was another hyperbolic navigation system, operational from the 1970s until 1997. It was the first truly global radio navigation system, with worldwide

coverage and continuous availability. However, the position accuracy was only a few kilometers [6].

VOR was launched as a U.S. standard in 1946 and subsequently adopted globally. Most VOR stations are co-sited with DME ranging stations and provide a service to aircraft over a range of up to 400 km. VOR/DME stations cover major airways over land, but coverage is nowhere near universal.

The other major terrestrial radio navigation system, debatably a guidance system, is ILS. This was also developed shortly after World War II and is used only for airport approach and landing.

These navigation systems do not offer the same coverage or accuracy as GNSS and are limited to providing horizontal position only. However, they provide a useful backup to GNSS for safety-critical applications.

Current developments in terrestrial radio navigation technology are focused on urban areas, where GNSS performance can be poor. Mobile phone and wireless local area network (WLAN) positioning is now established, while techniques based on TV and radio broadcasts, and ultra-wideband (UWB) communications have been prototyped. Terrestrial radio navigation systems are described in more detail in Chapter 9.

1.3.2 Satellite Navigation

The world's first satellite navigation system was the U.S. Navy's Transit system [6]. Development started in 1958, with the first experimental satellite launched in 1961 and system operational in 1964. The system was opened to civil use from 1967 and was decommissioned in 1996. Transit comprised between 4 and 7 low-altitude (1,100 km) satellites, each broadcasting at 150 and 400 MHz. No more than one satellite was in view at a time, and there was up to 100 minutes between satellite passes. Transit used Doppler positioning, which provided only one independent two-dimensional position fix per satellite pass. Consequently, it was of practical use only for shipping and geodesy applications. The single-pass position accuracy for stationary users was about 25m, while accuracy for moving users was degraded unless the Transit receiver was aided with dead-reckoning information. Russia developed and operated an almost identical system, known as Tsikada.

Development of the Global Positioning System (GPS) started in 1973 when a number of U.S. military satellite navigation programs were merged [8, 9]. The first operational prototype satellite was launched in 1978 and initial operational capability (IOC) of the full GPS system was declared in 1993. Although developed as a military system, GPS is now used for a wide range of civil applications. The Global Navigation Satellite System (GLONASS) is operated by Russia and was developed in parallel to GPS, also as a military system. The first satellite was launched in 1982. A third satellite navigation system, Galileo, is under development by the European Union and other partners. Galileo is a civil system under civil control. The first satellite launch was at the end of 2005 and IOC is planned for 2010–2012. In addition, regional systems are being developed by China, India, and Japan, with proposals to expand the Chinese Compass system to global coverage. GPS and GLONASS are currently undergoing major modernization programs.

These systems, collectively known as global navigation satellite systems, operate under the same principle.

GPS, GLONASS, and Galileo are each designed to comprise a constellation of 24 or more satellites orbiting at a radius of between 25,000 and 30,000 km, ensuring that signals from at least four satellites are available at any location. Each satellite broadcasts synchronized timing signals over two or three frequency bands between 1,145 and 1,614 MHz. A satellite navigation receiver may derive a three-dimensional position fix and calibrate its clock offset by passive ranging from four satellites. Figure 1.4 illustrates the basic concept. In practice, there are usually more satellites in view, enabling the position accuracy to be refined and consistency checks to be performed.

Each GNSS system broadcasts a range of different signals on a number of frequencies. Many signals are open to all users free of charge, whereas others are restricted to military users, emergency and security services, or commercial subscribers. These are augmented with additional information transmitted by space-based augmentation systems (SBAS), such as the Wide Area Augmentation System (WAAS) and European Geostationary Navigation Overlay Service (EGNOS).

In addition to GNSS, China currently operates a regional satellite navigation system, Beidou, based on two-way ranging.

The satellite navigation systems and services are described in detail in Chapter 6, while Chapter 7 describes how GNSS works, including the satellite orbits and signals, the receiver, and the navigation processor. Chapter 8 describes advanced GNSS technology, including use of differential and carrier phase techniques to improve precision, improving performance in poor signal-to-noise and multipath interference environments, and signal monitoring.

GNSS offers a basic radial positioning accuracy of 1.0–3.9m in the horizontal plane and 1.6–6.3m in the vertical axis, depending on the service, receiver design, and signal geometry. Differential techniques can improve this to within a meter by making use of base stations at known locations to calibrate some of the errors. Carrier-phase positioning can give centimeter accuracy for real-time navigation and millimeter accuracy for surveying and geodesy applications. It can also be used to measure attitude. However, carrier phase techniques are much more sensitive to interference, signal interruptions, and satellite geometry than basic positioning.

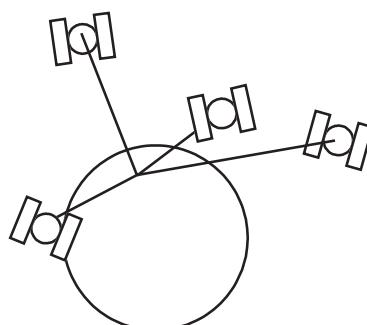


Figure 1.4 Ranging using four satellite navigation signals.

The principal advantages of GNSS are high long-term position accuracy and low-cost user equipment, from less than \$/€100 for a basic receiver. The main limitation of satellite navigation is lack of signal continuity. GNSS signals are vulnerable to interference, both incidental and deliberate. They can also be blocked, attenuated, and reflected by buildings, terrain, and foliage. For example, GNSS signals indoors and in the “urban canyons” between tall buildings are often weak and subject to multipath interference, which distorts the ranging measurements. In addition, the navigation solution bandwidth is relatively low (typically 10 Hz) and the short-term noise is high, compared to even low-grade inertial navigation systems, though this is not a problem for many applications.

1.4 Feature Matching

Feature-matching techniques determine the user’s position by comparing features of the terrain or environment with a database, much as a person would compare landmarks with a map or a set of directions.

Figure 1.5 illustrates the concept of terrain-referenced navigation for aircraft navigation. A radalt is used to measure the height of an aircraft above terrain, which is differenced with the vehicle height from the navigation solution to obtain the terrain height. By comparing a series of measurements with a terrain height database, the vehicle position may be determined, provided the terrain height varies sufficiently. Development of TRN started in the 1960s and a number of systems are now on the market.

TRN is accurate to about 50m. It works best over hilly and mountainous terrain and will not give position fixes over flat terrain or water. Limiting factors are the large footprint of the radalt sensor and the quality of the database. Replacing the radalt with a scanning laser rangefinder improves the resolution of the system, while sonar enables TRN to be performed underwater.

Aircraft image-matching navigation systems capture images of the terrain below using a camera, synthetic aperture radar (SAR), or a scanning laser system. Position fixes may then be obtained by extracting features from the images and comparing them with a database. Image matching tends to work better over flatter terrain, where there are more suitable features, so it is complementary to TRN. Accuracy

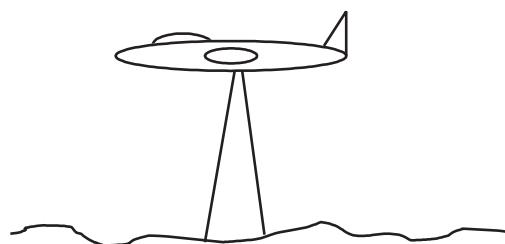


Figure 1.5 The concept of terrain-referenced navigation. (From: [4] © 2002 QinetiQ Ltd. Reprinted with permission.)

is of the order of 20m. Image matching may also be operated in a dead reckoning mode by comparing successive images to obtain velocity and angular rate.

Map-matching techniques use the fact that land vehicles generally travel on roads or rails and pedestrians do not walk through walls to constrain the drift of a dead reckoning solution and/or correct errors in a positioning measurement. They follow the navigation solution on a map and apply corrections where it strays outside the permitted areas.

Other feature-matching techniques include the measurement of anomalies in the Earth's magnetic or gravity field and stellar navigation. Feature matching is described in more detail in Chapter 11. Note that all feature-matching systems occasionally produce false fixes.

1.5 The Complete Navigation System

Different navigation applications have very different requirements in terms of accuracy, update rate, reliability, budget, size, and mass, and whether an attitude solution is required as well as position and velocity. For example, high-value, safety-critical assets, such as airliners and ships, require a guarantee that the navigation solution is always within the error bounds indicated and a high level of availability, but the accuracy requirements are relatively modest and there is a large budget. For military applications, a degree of risk is accepted, but the navigation system must be stealthy and able to operate in an electronic warfare environment; the accuracy requirements vary. For most personal navigation, road vehicle, and asset-tracking applications, the key drivers are cost, size, weight, and power consumption. Consequently, different combinations of navigation sensors are suitable for different applications.

Position fixing and dead reckoning systems have very different error characteristics so, for many applications, a dead reckoning system such as INS is integrated with one or more position fixing systems, such as GNSS. Figure 1.6 shows a typical integration architecture. The dead reckoning system provides the integrated navigation solution as it operates continuously, while measurements from the position-fixing system are used by an estimation algorithm to apply corrections to the dead reckoning system's navigation solution. The estimation algorithm is usually based on the Kalman filter, described in Chapter 3. INS/GNSS integration is described in Chapter 12, while multisensor integrated navigation is discussed in

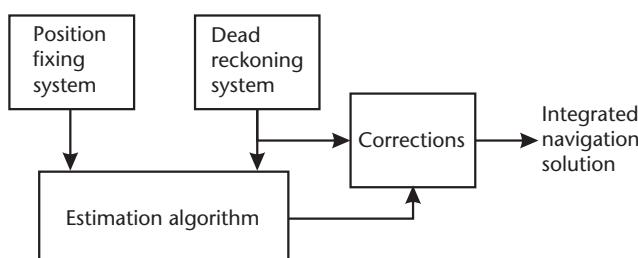


Figure 1.6 A typical integrated navigation architecture.

Chapter 14. Chapter 13 covers the related topics of INS calibration using transfer alignment, quasi-stationary alignment, and zero velocity updates.

To guarantee a reliable navigation solution, it is necessary to detect faults within the navigation system, whether they lie in the user equipment hardware, radio navigation signals, or within the estimation algorithm. This is known as integrity monitoring and can be provided at various levels. *Fault detection* simply informs the user that a fault is present; *fault isolation* identifies where the fault has occurred and produces a new the navigation solution without data from the faulty component; and *fault exclusion* additionally verifies that the new navigation solution is fault-free. Fault detection and integrity monitoring within the navigation user equipment is described in Chapter 15. However, faults in radio navigation signals can be more effectively detected by base stations at known locations, with alerts then transmitted to the navigation system user, as discussed in Section 8.5 for GNSS. For safety-critical applications, such as civil aviation, the integrity monitoring system must be formally certified to ensure it meets a number of performance requirements.

References

- [1] *The Concise Oxford Dictionary*, 9th ed., Oxford, U.K.: Oxford University Press, 1995.
- [2] Sobel, D., *Longitude*, London, U.K.: Fourth Estate, 1996.
- [3] Tull, W. J., “The Early History of Airborne Doppler Systems,” *Navigation: JION*, Vol. 43, No. 1, 1996, pp. 9–24.
- [4] Groves, P. D., “Principles of Integrated Navigation,” Course Notes, QinetiQ Ltd., 2002.
- [5] Goebel, G., *The Wizard War: WW2 and the Origins of Radar*, <http://www.vectorsite.net/ttwiza.html>, 1995.
- [6] Enge, P., et al., “Terrestrial Radionavigation Technologies,” *Navigation: JION*, Vol. 42, No. 1, 1995, pp. 61–108.
- [7] Stansell, T. A., Jr., “The Many Faces of Transit,” *Navigation: JION*, Vol. 25, No. 1, 1978, pp. 55–70.
- [8] Parkinson, B. W., “Origins, Evolution, and Future of Satellite Navigation,” *Journal of Guidance, Control and Dynamics*, Vol. 20, No. 1, 1997, pp. 11–25.
- [9] McDonald, K. D., “Early Development of the Global Positioning System,” in *Galileo: Europe’s Guiding Star*, W. Blanchard, (ed.), London, U.K.: Faircourt Ltd., 2006, pp. 114–128.

PART II

Navigation Mathematics

Coordinate Frames, Kinematics, and the Earth

This chapter provides the mathematical foundations for explaining the principles of navigation systems and their integration. Section 2.1 introduces the concept of a coordinate frame and how it may be used to represent an object, reference, or set of resolving axes. The main coordinate frames used in navigation are described. Section 2.2 explains the different methods of representing attitude and shows how to convert between them. It also defines the Cartesian position, velocity, acceleration, and angular rate in a multiple coordinate frame environment where the reference frame or resolving axes may be rotating. Section 2.3 shows how the Earth's surface is modeled and defines latitude, longitude, and height. It also introduces specific force and explains the difference between gravity and gravitation. Finally, Section 2.4 presents the equations for transforming between different coordinate frame representations.

2.1 Coordinate Frames

In simple mechanics problems, motion is modeled with respect to the Earth while pretending that the Earth is an inertial frame, ignoring its rotation. For navigation this does not work; the Earth's rotation has a significant impact on navigation computation as shown later. Navigation is also a multiple coordinate frame problem. Inertial sensors measure their motion with respect to an inertial frame. GPS measures the position and velocity of a receiver's antenna with respect to a constellation of satellites. However, the user wants to know their position with respect to the Earth.¹

Thus, for accurate navigation, the relationship between the different coordinate frames must be properly modeled. A convention is adopted here of using Greek letters to denote generic coordinate frames and Roman letters to denote specific frames.

A *coordinate frame* may be defined in two ways. It provides an origin and a set of axes in terms of which the motion of objects may be described (i.e., a reference). It also defines the position and orientation of an object. The two definitions are interchangeable. In a two-frame problem, defining which is the object frame and which is the reference frame is arbitrary. It is equally valid to describe the position and orientation of frame α with respect to frame β as it is to describe

frame β with respect to frame α . This is a principle of relativity: the laws of physics appear the same for all observers. In other words, describing the position of a road with respect to a car conveys the same information as the position of the car with respect to the road.²

An orthogonal coordinate frame has six degrees of freedom, the position of the origin, \mathbf{o} , and the orientation of the axes, x , y , and z . These must be expressed with respect to another frame in order to define them. Figure 2.1 illustrates this. Any navigation problem thus involves at least two coordinate frames: an object frame and a reference frame. The object frame describes the body whose position and/or orientation is desired, while the reference frame describes a known body, such as the Earth, relative to which the object position and/or orientation is desired. Many navigation problems involve more than one reference frame or even more than one object frame.

All coordinate frames considered here form *orthogonal right-handed* basis sets. This means that the x -, y -, and z -axes are always mutually perpendicular and are oriented such that if the thumb and first two fingers of the right hand are extended perpendicularly, the thumb is the x -axis, the first finger is the y -axis, and the second finger is the z -axis.¹

Any two coordinate frames may have any relative position and attitude. They may also have any relative velocity, acceleration, rotation, and so forth. The orientation of one frame with respect to another is a unique set of numbers, though there is more than one way of representing attitude. However, the other kinematic quantities are not. They comprise vectors, which may be resolved into components along any three mutually perpendicular axes. For example, the position of frame α with respect to frame β may be described using the α -frame axes, the β -frame axes, or the axes of a third frame, γ . Here, a superscript is used to denote the axes in which a quantity is expressed,² known as the resolving frame. Note that it is not necessary to define the origin of the resolving frame.

The attitude, position, velocity, acceleration, and angular rate in a multiple coordinate frame problem are defined in Section 2.2. The remainder of this section defines the main coordinate frames used in navigation problems: the Earth-centered inertial (ECI), Earth-centered Earth-fixed (ECEF), local navigation, and body frames. A brief summary of other coordinate frames used in navigation completes the section.

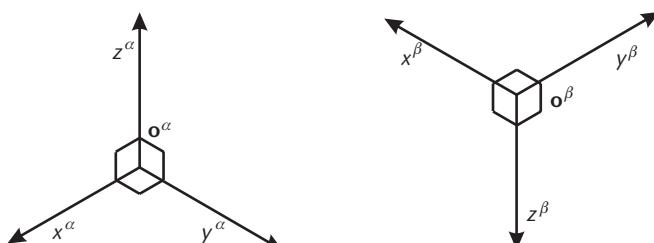


Figure 2.1 Two orthogonal coordinate frames. (From: [1]. © 2002 QinetiQ Ltd. Reprinted with permission.)

2.1.1 Earth-Centered Inertial Frame

In physics, an *inertial* coordinate frame is one that does not accelerate or rotate with respect to the rest of the Universe. This does not define a unique coordinate frame. In navigation, a more specific form of the inertial frame, known as the *Earth-centered inertial* frame, is used, denoted by the symbol i . This is nominally centered at the Earth's center of mass and oriented with respect to the Earth's spin axis and the stars. Strictly, this is not a true inertial frame, as the Earth experiences acceleration in its orbit around the Sun, its spin axis slowly moves, and the Galaxy rotates. However, it is a sufficiently accurate approximation to an inertial frame for navigation purposes.¹

Figure 2.2 shows the axes of the ECI frame. The rotation shown is that of the Earth with respect to space. The z -axis always points along the Earth's axis of rotation from the center to the north pole (true, not magnetic). The x - and y -axes lie within the equatorial plane. They do not rotate with the Earth, but the y -axis always lies 90 degrees ahead of the x -axis in the direction of rotation. This does not uniquely define the coordinate frame; it is also necessary to specify the time at which the inertial frame axes coincide with those of the ECEF frame (see Section 2.1.2).² There are two common solutions. The first is simply to align the two coordinate frames when the navigation solution is initialized. The other option, used within the scientific community, is to define the x -axis as the direction from the Earth to the Sun at the vernal equinox, which is the spring equinox in the northern hemisphere. This is the same as the direction from the center of the Earth to the intersection of the Earth's equatorial plane with the Earth-Sun orbital plane (ecliptic).

A further problem, where a precise definition of the coordinate frame is needed, is polar motion. The spin axis actually moves with respect to the solid Earth, with the poles roughly following a circular path of radius 15m. One solution is to adopt the IERS reference pole (IRP) or conventional terrestrial pole (CTP), which is the average position of the pole surveyed between 1900 and 1905. The version of the inertial frame that adopts the IRP/CTP, the Earth's center of mass as its origin, and the x -axis based on the Earth-Sun axis at vernal equinox is known as the conventional inertial reference system (CIRS).

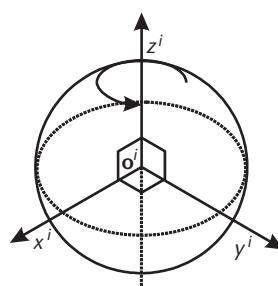


Figure 2.2 Axes of the ECI frame. (From: [1]. © 2002 QinetiQ Ltd. Reprinted with permission.)

The inertial frame is important in navigation because inertial sensors measure motion with respect to a generic inertial frame, and it enables the simplest form of navigation equations to be used, as shown in later chapters.

2.1.2 Earth-Centered Earth-Fixed Frame

The *Earth-centered Earth-fixed* frame, commonly abbreviated to Earth frame, is similar to the ECI frame, except that all axes remain fixed with respect to the Earth. The ECEF frame is denoted by the symbol e and has its origin at the center of the ellipsoid modeling the Earth's surface, which is roughly at the center of mass.

Figure 2.3 shows the axes of the ECEF frame. The z -axis always points along the Earth's axis of rotation from the center to the North Pole (true, not magnetic). The x -axis points from the center to the intersection of the equator with the IERS reference meridian (IRM) or conventional zero meridian (CZM), which defines 0 degree longitude. The y -axis completes the right-handed orthogonal set, pointing from the center to the intersection of the equator with the 90-degree east meridian. The ECEF frame using the IRP/CTP and the IRM/CZM is also known as the conventional terrestrial reference system (CTRS), and some authors use the symbol t to denote it.

The Earth frame is important in navigation because the user wants to know their position relative to the Earth, so it is commonly used as both a reference frame and a resolving frame.

2.1.3 Local Navigation Frame

The *local navigation frame*, local level navigation frame, geodetic, or geographic frame is denoted by the symbol n (some authors use g). Its origin is the point a navigation solution is sought for (i.e., the navigation system, the user, or the host vehicle's center of mass).¹

Figure 2.4 shows the axes of the local navigation frame. The z axis, also known as the down (D) axis, is defined as the normal to the surface of the reference ellipsoid (Section 2.3.1), pointing roughly toward the center of the Earth. Simple gravity models (see Section 2.3.5) assume that the gravity vector is coincident with

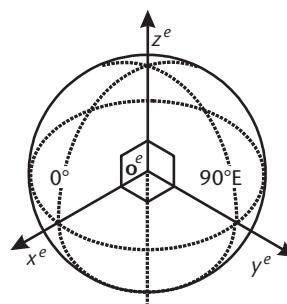


Figure 2.3 Axes of the ECEF frame. (From: [1]. © 2002 QinetiQ Ltd. Reprinted with permission.)

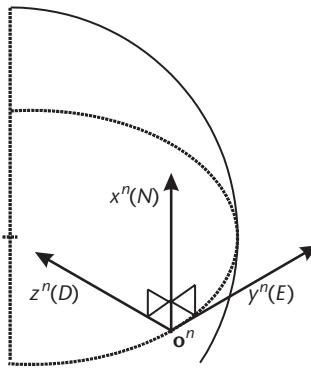


Figure 2.4 Axes of the local navigation frame. (From: [1]. © 2002 QinetiQ Ltd. Reprinted with permission.)

the z axis of the local navigation frame. True gravity deviates from this slightly due to local anomalies. The x -axis, or north (N) axis, is the projection in the plane orthogonal to the z -axis of the line from the user to the north pole. By completing the orthogonal set, the y -axis always points east and is hence known as the east (E) axis.

North, east, down is the most common form of the local navigation frame and will always be used here. However, there are other forms in use, such as x = east, y = north, z = up, and x = south, y = west, z = down.²

The local navigation frame is important in navigation because the user wants to know their attitude relative to the north, east, and down directions. For position and velocity, it provides a convenient set of resolving axes, but is not used as a reference frame.

A major drawback of the local navigation frame is that there is a singularity at each pole because the north and east axes are undefined there. Thus, navigation equations mechanized using this frame are unsuitable for use near the poles. Instead, an alternative frame should be used with conversion of the navigation solution to the local navigation frame at the end of the processing chain.

In a multibody problem, there are several local navigation frames in play. However, in practice, only one tends to be of interest. Also, the differences in orientation between the local navigation frames of objects within a few meters can usually be neglected.³

2.1.4 Body Frame

The *body frame*, sometimes known as the vehicle frame, comprises the origin and orientation of the object for which a navigation solution is sought. The origin is coincident with that of the local navigation frame, but the axes remain fixed with respect to the body and are generally defined as x = forward (i.e., the usual direction of travel), z = down (i.e., the usual direction of gravity), and y = right, completing the orthogonal set. For angular motion, the x -axis is the roll axis, the y -axis is the pitch axis, and the z -axis is the yaw axis. Hence, the axes of the body frame are sometimes known as roll, pitch, and yaw. Figure 2.5 illustrates this. A right-handed

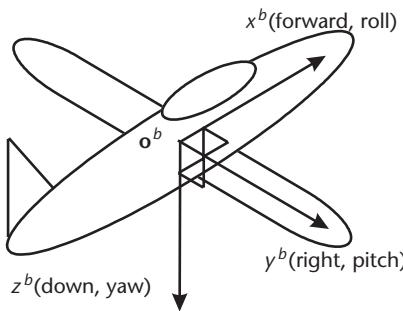


Figure 2.5 Body frame axes. (From: [1]. © 2002 QinetiQ Ltd. Reprinted with permission.)

corkscrew rule applies, whereby if the axis is pointing away, then positive rotation about that axis is clockwise.¹

The body frame is essential in navigation because it describes the object that is navigating. All strapdown inertial sensors measure the motion of the body frame (with respect to a generic inertial frame).

The symbol b is used to denote the body frame of the primary object of interest. However, many navigation problems involve multiple objects, each with their own body frame, for which alternative symbols, such as s for a satellite and a for an antenna, must be used.

2.1.5 Other Frames

The *geocentric frame*, denoted c , is similar to the local navigation frame, except that the z axis points from the origin to the center of the Earth. The x -axis is again the projection of the line to the north pole in the orthogonal plane to the z -axis.¹

The *tangent plane frame*, t , is also known as the local geodetic frame. It is similar to the local navigation frame, except that the origin does not coincide with the navigating object but is instead fixed relative to the Earth. This frame is used for navigation within a localized area, such as aircraft landing.

The origin and z -axis of the *wander azimuth frame*, w (some authors use n), are coincident with that of the local navigation frame. However, the x and y axes are displaced from north and east by an angle, ψ_{nv} or α , known as the wander angle, that varies as the frame moves with respect to the Earth. Use of this frame avoids the polar singularity of the local navigation frame, so it is commonly used to mechanize inertial navigation equations. The wander angle is always known, so transformation of the navigation solution to the local navigation frame is straightforward. Wander-azimuth navigation equations are summarized in Section 5.3.5.

In most systems, the sensitive axes of the inertial instruments are nominally aligned with the body frame axes, but there will always be some deviation. Therefore, some authors like to treat these sensitive axes as separate *inertial instrument frames*, with a coordinate frame for each accelerometer and gyro. However, it is generally simpler to treat the departures from the body axes (i.e., the instrument mounting misalignments), as a set of perturbations. Some IMUs are manufactured in a “skew” configuration, whereby the sensors’ sensitive axes are not aligned with

the casing. In this case, the inertial instrument frame must be considered distinct from the body frame, though the transformation is often performed within the IMU.

A further issue is that, because of the finite size of the sensors, the origin of each sensor will be slightly different. Prior to computation of the navigation solution, the inertial sensor outputs must be transformed to a common point of reference. Again, this transformation is usually performed within the IMU.²

In calculating the motion of GNSS satellites, orbital coordinate frames, denoted by α , are used. These are defined in Section 7.1.1.

2.2 Kinematics

In navigation, the linear and angular motion of one coordinate frame must be described with respect to another. Most kinematic quantities, such as position, velocity, acceleration, and angular rate, involve three coordinate frames:

- The frame whose motion is described, known as the *object frame*, α ;
- The frame with which that motion is respect to, known as the *reference frame*, β ;
- The set of axes in which that motion is represented, known as the *resolving frame*, γ .

The object frame, α , and the reference frame, β , must be different; otherwise, there is no motion. The resolving frame, γ , may be the object frame, the reference frame, or a third frame. To describe these kinematic quantities fully, all three frames must be explicitly stated. Most authors do not do this, potentially causing confusion. Here, the following notation is used for Cartesian position, velocity, acceleration, and angular rate:

$$\mathbf{x}_{\beta\alpha}^{\gamma}$$

where the vector, \mathbf{x} , describes a kinematic property of frame α with respect to frame β , expressed in the frame γ axes. For attitude, only the object frame, α , and reference frame, β , are involved; there is no resolving frame.

This section begins by describing the different forms of attitude representation: Euler angles, coordinate transformation matrices, quaternion attitude, and rotation vectors. All methods of representing attitude fulfill two functions. They describe the orientation of one coordinate frame with respect to another (e.g., an object frame with respect to a reference frame). They also provide a means of transforming a vector from one set of resolving axes to another.

Later, the angular rate, Cartesian (as opposed to curvilinear) position, velocity, and acceleration are described. In defining the velocity and acceleration, it is important to correctly account for any rotation of the reference frame and resolving frame.

2.2.1 Euler Attitude

Euler angles (pronounced “Oiler”) are the most intuitive way of describing an attitude, particularly that of a body frame with respect to the corresponding local navigation frame. The attitude is broken down into three successive rotations. This can be illustrated by the transformation of a vector, $\mathbf{x} = (x, y, z)$, from one set of resolving axes, β , to a second set, α . Figure 2.6 shows the three rotations that comprise the transformation.

The first rotation, $\psi_{\beta\alpha}$, is the *yaw* rotation. This is performed about the common z -axis of the β frame and the first intermediate frame. It transforms the x and y components of the vector, but leaves the z component unchanged. The resulting vector is resolved about the axes of the first intermediate frame, denoted by the superscript, ψ :

$$\begin{aligned} x^\psi &= x^\beta \cos \psi_{\beta\alpha} + y^\beta \sin \psi_{\beta\alpha} \\ y^\psi &= -x^\beta \sin \psi_{\beta\alpha} + y^\beta \cos \psi_{\beta\alpha} \\ z^\psi &= z^\beta \end{aligned} \quad (2.1)$$

Next, the *pitch* rotation, $\theta_{\beta\alpha}$, is performed about the common y -axis of the first and second intermediate frames. Here, the x and z components of the vector are transformed, resulting in a vector resolved about the axes of the second intermediate frame, denoted by the superscript, θ :

$$\begin{aligned} x^\theta &= x^\psi \cos \theta_{\beta\alpha} - z^\psi \sin \theta_{\beta\alpha} \\ y^\theta &= y^\psi \\ z^\theta &= x^\psi \sin \theta_{\beta\alpha} + z^\psi \cos \theta_{\beta\alpha} \end{aligned} \quad (2.2)$$

Finally, the *roll* rotation, $\phi_{\beta\alpha}$, is performed about the common x -axis of the second intermediate frame and the α frame, transforming the y and z components and leaving the vector resolved about the axes of the α frame:

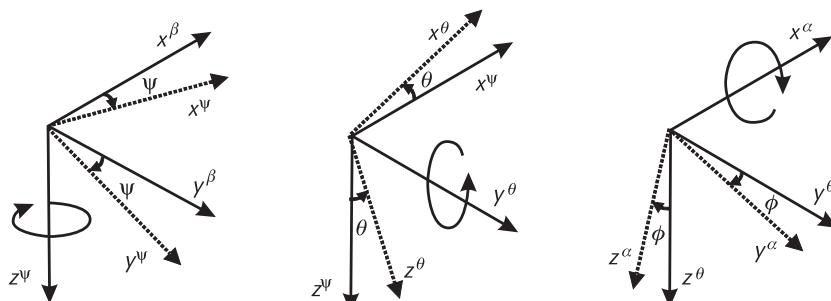


Figure 2.6 Euler angle rotations.

$$\begin{aligned}x^\alpha &= x^\theta \\y^\alpha &= y^\theta \cos \phi_{\beta\alpha} + z^\theta \sin \phi_{\beta\alpha} \\z^\alpha &= -y^\theta \sin \phi_{\beta\alpha} + z^\theta \cos \phi_{\beta\alpha}\end{aligned}\quad (2.3)$$

Although it is easier to illustrate the Euler angles in terms of transforming the resolving axes of a vector, the roll, pitch, and yaw rotations, $\phi_{\beta\alpha}$, $\theta_{\beta\alpha}$, and $\psi_{\beta\alpha}$, equally describe the orientation of the object frame, α , with respect to the reference frame, β . In the specific case where the Euler angles describe the attitude of the body frame with respect to the local navigation frame, the roll rotation, ϕ_{nb} , is known as *bank*, the pitch rotation, θ_{nb} , is known as *elevation*, and the yaw rotation, ψ_{nb} , is known as *heading* or *azimuth*. Some authors use the term attitude to describe only the bank and elevation, excluding heading. The bank and elevation are also collectively known as tilts. Here, attitude always describes all three components of orientation.

The Euler rotation from frame β to frame α may be denoted by the vector

$$\boldsymbol{\psi}_{\beta\alpha} = \begin{pmatrix} \phi_{\beta\alpha} \\ \theta_{\beta\alpha} \\ \psi_{\beta\alpha} \end{pmatrix} \quad (2.4)$$

noting that the Euler angles are listed in the reverse order to that in which they are applied. The order in which the three rotations are carried out is critical. If they are performed in a different order (e.g., with the roll first), the orientation of the axes at the end of the transformation is generally different. In formal terms, the three Euler rotations do not commute.

The Euler rotation $(\phi_{\beta\alpha} + \pi, \pi - \theta_{\beta\alpha}, \psi_{\beta\alpha} + \pi)$ gives the same result as the Euler rotation $(\phi_{\beta\alpha}, \theta_{\beta\alpha}, \psi_{\beta\alpha})$. Consequently, to avoid duplicate sets of Euler angles representing the same attitude, a convention is adopted of limiting the pitch rotation, θ , to the range $-90^\circ \leq \theta \leq 90^\circ$. Another property of Euler angles is that the axes about which the roll and yaw rotations are made are usually not orthogonal, although both are orthogonal to the axis about which the pitch rotation is made.

To reverse an Euler rotation, either the original operation must be reversed, beginning with the roll, or a different transformation must be applied. Simply reversing the sign of the Euler angles does not return to the original orientation, thus¹

$$\begin{pmatrix} \phi_{\alpha\beta} \\ \theta_{\alpha\beta} \\ \psi_{\alpha\beta} \end{pmatrix} \neq \begin{pmatrix} -\phi_{\beta\alpha} \\ -\theta_{\beta\alpha} \\ -\psi_{\beta\alpha} \end{pmatrix} \quad (2.5)$$

Similarly, successive rotations cannot be expressed simply by adding the Euler angles:

$$\begin{pmatrix} \phi_{\beta\gamma} \\ \theta_{\beta\gamma} \\ \psi_{\beta\gamma} \end{pmatrix} \neq \begin{pmatrix} \phi_{\beta\alpha} + \phi_{\alpha\gamma} \\ \theta_{\beta\alpha} + \theta_{\alpha\gamma} \\ \psi_{\beta\alpha} + \psi_{\alpha\gamma} \end{pmatrix} \quad (2.6)$$

A further difficulty is that the Euler angles exhibit a singularity at $\pm 90^\circ$ pitch, where the roll and yaw become indistinguishable. Because of these difficulties, Euler angles are rarely used for attitude computation.²

2.2.2 Coordinate Transformation Matrix

The *coordinate transformation matrix* is a 3×3 matrix, denoted C_α^β (some authors use R or T). Where it is used to transform a vector from one set of resolving axes to another, the lower index represents the “from” coordinate frame and the upper index the “to” frame. The rows of a coordinate transformation matrix are in the “to” frame, whereas the columns are in the “from” frame. Thus,

$$\mathbf{x}_{\delta\gamma}^\beta = C_\alpha^\beta \mathbf{x}_{\delta\gamma}^\alpha \quad (2.7)$$

where \mathbf{x} is any quantity. Where the matrix is used to represent attitude, it is more common to use the upper index to represent the reference frame, β , and the lower index to represent the object frame, α . Hence the matrix is representing the rotation from the object frame to the reference frame, the opposite convention to that used for Euler angles. However, it is equally valid to represent attitude as a reference frame to object frame transformation, C_β^α .

Figure 2.7 shows the role of each element of the coordinate transformation matrix in transforming a vector from frame α to frame β resolving axes. Rearranging (2.7), the coordinate transformation matrix can be obtained from the product of any vector expressed in the two frames:

$$C_\alpha^\beta = \frac{\mathbf{x}_{\delta\gamma}^\beta \mathbf{x}_{\delta\gamma}^{\alpha T}}{|\mathbf{x}_{\delta\gamma}|^2} \quad (2.8)$$

It can be shown [2] that the coordinate transformation matrix elements are the product of the unit vectors describing the axes of the two frames, which, in turn, are equal to the cosines of the angles between the axes:

$\alpha_x > \beta_x$	$\alpha_y > \beta_x$	$\alpha_z > \beta_x$
$\alpha_x > \beta_y$	$\alpha_y > \beta_y$	$\alpha_z > \beta_y$
$\alpha_x > \beta_z$	$\alpha_y > \beta_z$	$\alpha_z > \beta_z$

Figure 2.7 The coordinate transformation matrix component functions.

$$\mathbf{C}_\alpha^\beta = \begin{pmatrix} \mathbf{u}_{\beta x} \cdot \mathbf{u}_{\alpha x} & \mathbf{u}_{\beta x} \cdot \mathbf{u}_{\alpha y} & \mathbf{u}_{\beta x} \cdot \mathbf{u}_{\alpha z} \\ \mathbf{u}_{\beta y} \cdot \mathbf{u}_{\alpha x} & \mathbf{u}_{\beta y} \cdot \mathbf{u}_{\alpha y} & \mathbf{u}_{\beta y} \cdot \mathbf{u}_{\alpha z} \\ \mathbf{u}_{\beta z} \cdot \mathbf{u}_{\alpha x} & \mathbf{u}_{\beta z} \cdot \mathbf{u}_{\alpha y} & \mathbf{u}_{\beta z} \cdot \mathbf{u}_{\alpha z} \end{pmatrix} = \begin{pmatrix} \cos \mu_{\beta x, \alpha x} & \cos \mu_{\beta x, \alpha y} & \cos \mu_{\beta x, \alpha z} \\ \cos \mu_{\beta y, \alpha x} & \cos \mu_{\beta y, \alpha y} & \cos \mu_{\beta y, \alpha z} \\ \cos \mu_{\beta z, \alpha x} & \cos \mu_{\beta z, \alpha y} & \cos \mu_{\beta z, \alpha z} \end{pmatrix} \quad (2.9)$$

where \mathbf{u}_i is the unit vector describing axis i and $\mu_{i,j}$ is the resultant angle between axes i and j . Hence, the term direction cosine matrix (DCM) is often used to describe these matrices.

Coordinate transformation matrices are easy to manipulate. As (2.9) shows, to reverse a rotation or coordinate transformation, the transpose of the matrix, denoted by the superscript, T (see Section A.2 in Appendix A), is used. Thus,

$$\mathbf{C}_\beta^\alpha = (\mathbf{C}_\alpha^\beta)^T \quad (2.10)$$

To perform successive transformations or rotations, the coordinate transformation matrices are simply multiplied:

$$\mathbf{C}_\alpha^\gamma = \mathbf{C}_\beta^\gamma \mathbf{C}_\alpha^\beta \quad (2.11)$$

However, as with any matrix multiplication, the order is critical, so

$$\mathbf{C}_\alpha^\gamma \neq \mathbf{C}_\alpha^\beta \mathbf{C}_\beta^\gamma \quad (2.12)$$

Performing a transformation and then reversing the process must return the original vector or matrix, so

$$\mathbf{C}_\alpha^\beta \mathbf{C}_\beta^\alpha = \mathbf{I}_3 \quad (2.13)$$

where \mathbf{I}_n is the $n \times n$ identity or unit matrix. Thus, coordinate transformation matrices are orthonormal (see Section A.3 in Appendix A).

Although a coordinate transformation matrix has nine components, the requirement to meet (2.13) means that only three of these are independent. Thus, it has the same number of independent components as Euler attitude. A set of Euler angles is converted to a coordinate transformation matrix by first representing each of the rotations of (2.1)–(2.3) as a matrix and then multiplying, noting that with matrices, the first operation is placed on the right. Thus,

$$\begin{aligned}
C_{\beta}^{\alpha} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_{\beta\alpha} & \sin \phi_{\beta\alpha} \\ 0 & -\sin \phi_{\beta\alpha} & \cos \phi_{\beta\alpha} \end{pmatrix} \begin{pmatrix} \cos \theta_{\beta\alpha} & 0 & -\sin \theta_{\beta\alpha} \\ 0 & 1 & 0 \\ \sin \theta_{\beta\alpha} & 0 & \cos \theta_{\beta\alpha} \end{pmatrix} \begin{pmatrix} \cos \psi_{\beta\alpha} & \sin \psi_{\beta\alpha} & 0 \\ -\sin \psi_{\beta\alpha} & \cos \psi_{\beta\alpha} & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
&= \begin{bmatrix} \cos \theta_{\beta\alpha} \cos \psi_{\beta\alpha} & \cos \theta_{\beta\alpha} \sin \psi_{\beta\alpha} & -\sin \theta_{\beta\alpha} \\ -\cos \phi_{\beta\alpha} \sin \psi_{\beta\alpha} & \cos \phi_{\beta\alpha} \cos \psi_{\beta\alpha} & \sin \phi_{\beta\alpha} \cos \theta_{\beta\alpha} \\ +\sin \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \cos \psi_{\beta\alpha} & +\sin \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \sin \psi_{\beta\alpha} & \cos \phi_{\beta\alpha} \cos \theta_{\beta\alpha} \\ \sin \phi_{\beta\alpha} \sin \psi_{\beta\alpha} & -\sin \phi_{\beta\alpha} \cos \psi_{\beta\alpha} & \cos \phi_{\beta\alpha} \cos \theta_{\beta\alpha} \\ +\cos \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \cos \psi_{\beta\alpha} & +\cos \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \sin \psi_{\beta\alpha} & \end{bmatrix} \quad (2.14)
\end{aligned}$$

and

$$C_{\alpha}^{\beta} = \begin{bmatrix} \cos \theta_{\beta\alpha} \cos \psi_{\beta\alpha} & \begin{pmatrix} -\cos \phi_{\beta\alpha} \sin \psi_{\beta\alpha} \\ +\sin \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \cos \psi_{\beta\alpha} \end{pmatrix} & \begin{pmatrix} \sin \phi_{\beta\alpha} \sin \psi_{\beta\alpha} \\ +\cos \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \cos \psi_{\beta\alpha} \end{pmatrix} \\ \cos \theta_{\beta\alpha} \sin \psi_{\beta\alpha} & \begin{pmatrix} \cos \phi_{\beta\alpha} \cos \psi_{\beta\alpha} \\ +\sin \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \sin \psi_{\beta\alpha} \end{pmatrix} & \begin{pmatrix} -\sin \phi_{\beta\alpha} \cos \psi_{\beta\alpha} \\ +\cos \phi_{\beta\alpha} \sin \theta_{\beta\alpha} \sin \psi_{\beta\alpha} \end{pmatrix} \\ -\sin \theta_{\beta\alpha} & \sin \phi_{\beta\alpha} \cos \theta_{\beta\alpha} & \cos \phi_{\beta\alpha} \cos \theta_{\beta\alpha} \end{bmatrix} \quad (2.15)$$

It is useful to define the operator $C(\)$ such that¹

$$C_{\beta}^{\alpha} = C(\psi_{\beta\alpha}) \quad (2.16)$$

The reverse transformation is

$$\begin{aligned}
\phi_{\beta\alpha} &= \arctan2(C_{\beta 2,3}^{\alpha}, C_{\beta 3,3}^{\alpha}) = \arctan2(C_{\alpha 3,2}^{\beta}, C_{\alpha 3,3}^{\beta}) \\
\theta_{\beta\alpha} &= -\arcsin C_{\beta 1,3}^{\alpha} = -\arcsin C_{\alpha 3,1}^{\beta} \\
\psi_{\beta\alpha} &= \arctan2(C_{\beta 1,2}^{\alpha}, C_{\beta 1,1}^{\alpha}) = \arctan2(C_{\alpha 2,1}^{\beta}, C_{\alpha 1,1}^{\beta})
\end{aligned} \quad (2.17)$$

noting that four-quadrant (360°) arctangent functions must be used where $\arctan2(a, b)$ is equivalent to $\arctan(a/b)$. It is useful to define the operator $\psi(\)$ such that

$$\psi_{\beta\alpha} = \psi(C_{\beta}^{\alpha}) \quad (2.18)$$

Where the coordinate transformation matrix and Euler angles represent a small angular perturbation for which the small angle approximation is valid, (2.14) becomes

$$\mathbf{C}_\beta^\alpha = \begin{pmatrix} 1 & \psi_{\beta\alpha} & -\theta_{\beta\alpha} \\ -\psi_{\beta\alpha} & 1 & \phi_{\beta\alpha} \\ \theta_{\beta\alpha} & -\phi_{\beta\alpha} & 1 \end{pmatrix} = \mathbf{I}_3 - [\boldsymbol{\psi}_{\beta\alpha} \wedge] \quad (2.19)$$

where $[\mathbf{x} \wedge]$ denotes the skew-symmetric matrix of \mathbf{x} (see Section A.3 in Appendix A).²

2.2.3 Quaternion Attitude

A rotation may be represented using a *quaternion*, which is a hyper-complex number with four components:¹

$$\mathbf{q} = (q_0, q_1, q_2, q_3)$$

where q_0 represents the magnitude of the rotation, and the other three components represent the axis about which that rotation takes place. Some authors place q_0 at the end and call it q_4 .²

Note that the axis of rotation is the same in both coordinate frames. It is the only vector that is invariant to the coordinate transformation. As with the other attitude representations, only three components are independent. The quaternion attitude is defined as

$$\mathbf{q}_{\beta\alpha} = \begin{pmatrix} \cos(\mu_{\beta,\alpha}/2) \\ \mathbf{u}_{\beta\alpha}^{\alpha/\beta} \sin(\mu_{\beta,\alpha}/2) \\ \mathbf{u}_{\beta\alpha}^{\alpha/\beta} \sin(\mu_{\beta,\alpha}/2) \\ \mathbf{u}_{\beta\alpha}^{\alpha/\beta} \sin(\mu_{\beta,\alpha}/2) \end{pmatrix} \quad (2.20)$$

where $\mu_{\beta,\alpha}$ is the rotation angle and $\mathbf{u}_{\beta\alpha}^{\alpha/\beta}$ is the unit vector of the rotation axis.

With only four components, quaternion attitude representation is computationally efficient. However, manipulation of quaternions is not intuitive, so their use in place of coordinate transformation matrices makes navigation equations more difficult to follow, increasing the chances of mistakes being made. Consequently, discussion of quaternions here is limited to their transformation to and from coordinate transformation matrices.¹

Quaternions are converted to a coordinate transformation matrix using

$$\begin{aligned} \mathbf{C}_\beta^\alpha &= \mathbf{C}(\mathbf{q}_{\beta\alpha}) \\ &= \begin{pmatrix} q_{\beta\alpha 0}^2 + q_{\beta\alpha 1}^2 - q_{\beta\alpha 2}^2 - q_{\beta\alpha 3}^2 & 2(q_{\beta\alpha 1}q_{\beta\alpha 2} - q_{\beta\alpha 3}q_{\beta\alpha 0}) & 2(q_{\beta\alpha 1}q_{\beta\alpha 3} + q_{\beta\alpha 2}q_{\beta\alpha 0}) \\ 2(q_{\beta\alpha 1}q_{\beta\alpha 2} + q_{\beta\alpha 3}q_{\beta\alpha 0}) & q_{\beta\alpha 0}^2 - q_{\beta\alpha 1}^2 + q_{\beta\alpha 2}^2 - q_{\beta\alpha 3}^2 & 2(q_{\beta\alpha 2}q_{\beta\alpha 3} - q_{\beta\alpha 1}q_{\beta\alpha 0}) \\ 2(q_{\beta\alpha 1}q_{\beta\alpha 3} - q_{\beta\alpha 2}q_{\beta\alpha 0}) & 2(q_{\beta\alpha 2}q_{\beta\alpha 3} + q_{\beta\alpha 1}q_{\beta\alpha 0}) & q_{\beta\alpha 0}^2 - q_{\beta\alpha 1}^2 - q_{\beta\alpha 2}^2 + q_{\beta\alpha 3}^2 \end{pmatrix} \end{aligned} \quad (2.21)$$

The reverse transformation is²

$$\begin{aligned}
q_{\beta\alpha 0} &= \frac{1}{2} \sqrt{1 + C_{\beta 1, 1}^{\alpha} + C_{\beta 2, 2}^{\alpha} + C_{\beta 3, 3}^{\alpha}} = \frac{1}{2} \sqrt{1 + C_{\alpha 1, 1}^{\beta} + C_{\alpha 2, 2}^{\beta} + C_{\alpha 3, 3}^{\beta}} \\
q_{\beta\alpha 1} &= \frac{C_{\beta 3, 2}^{\alpha} - C_{\beta 2, 3}^{\alpha}}{4q_{\beta\alpha 0}} &= \frac{C_{\alpha 2, 3}^{\beta} - C_{\alpha 3, 2}^{\beta}}{4q_{\beta\alpha 0}} \\
q_{\beta\alpha 2} &= \frac{C_{\beta 1, 3}^{\alpha} - C_{\beta 3, 1}^{\alpha}}{4q_{\beta\alpha 0}} &= \frac{C_{\alpha 3, 1}^{\beta} - C_{\alpha 1, 3}^{\beta}}{4q_{\beta\alpha 0}} \\
q_{\beta\alpha 3} &= \frac{C_{\beta 2, 1}^{\alpha} - C_{\beta 1, 2}^{\alpha}}{4q_{\beta\alpha 0}} &= \frac{C_{\alpha 1, 2}^{\beta} - C_{\alpha 2, 1}^{\beta}}{4q_{\beta\alpha 0}} \\
\mathbf{q}_{\beta\alpha} &= \mathbf{q}(\mathbf{C}_{\beta}^{\alpha})
\end{aligned} \tag{2.22}$$

More details on quaternion methods may be found in other navigation texts [3, 4].

2.2.4 Rotation Vector

The final method of representing attitude is the rotation vector [5]. This is a three-component vector, $\boldsymbol{\rho}$ (some authors use $\boldsymbol{\sigma}$), the direction of which gives the axis of rotation and the magnitude the angle of rotation. As with quaternions, the axis of rotation is the same in both coordinate frames. Like quaternions, manipulation of rotation vectors is not intuitive, so they will not be covered further here. A rotation vector is converted to a coordinate transformation matrix using

$$\mathbf{C}_{\beta}^{\alpha} = \mathbf{I}_3 - \frac{\sin|\boldsymbol{\rho}_{\beta\alpha}|}{|\boldsymbol{\rho}_{\beta\alpha}|} [\boldsymbol{\rho}_{\beta\alpha} \wedge] + \frac{1 - \cos|\boldsymbol{\rho}_{\beta\alpha}|}{|\boldsymbol{\rho}_{\beta\alpha}|^2} [\boldsymbol{\rho}_{\beta\alpha} \wedge]^2 \tag{2.23}$$

The reverse transformation is

$$\boldsymbol{\rho}_{\beta\alpha} = \frac{\theta}{2 \sin \theta} \begin{pmatrix} C_{\beta 2, 3}^{\alpha} - C_{\beta 3, 2}^{\alpha} \\ C_{\beta 3, 1}^{\alpha} - C_{\beta 1, 3}^{\alpha} \\ C_{\beta 1, 2}^{\alpha} - C_{\beta 2, 1}^{\alpha} \end{pmatrix}, \quad \theta = \arccos \left[\frac{\text{tr}(\mathbf{C}_{\beta}^{\alpha}) - 1}{2} \right] \tag{2.24}$$

where $\text{tr}(\cdot)$ is the trace of a matrix (see Section A.2 in Appendix A). In the small angle approximation, the rotation vector is the same as the Euler angles [see (2.19)]. Rotation vectors are useful for interpolating attitudes as the vector is simply scaled according to the interpolation point. More details on rotation vector methods may be found in [2].

2.2.5 Angular Rate

The *angular rate* vector, $\boldsymbol{\omega}_{\beta\alpha}^{\gamma}$, is the rate of rotation of the α -frame axes with respect to the β -frame axes, resolved about the γ -frame axes. Figure 2.8 illustrates the directions of the angular rate vector and the corresponding rotation. The

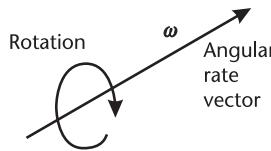


Figure 2.8 Angular rate rotation and vector directions.

rotation is within the plane perpendicular to the angular rate vector. The resolving axes may be changed simply by premultiplying by the relevant coordinate transformation matrix. Angular rates resolved about the same axes may simply be added, provided the object frame of one angular rate is the same as the reference frame of the other; thus³

$$\boldsymbol{\omega}_{\beta\alpha}^{\gamma} = \boldsymbol{\omega}_{\beta\delta}^{\gamma} + \boldsymbol{\omega}_{\delta\alpha}^{\gamma} \quad (2.25)$$

Some authors use the notation p , q , and r to denote the components of angular rate in, respectively, the x , y , and z axes of the resolving frame, so $\boldsymbol{\omega}_{\beta\alpha}^{\gamma} = (p_{\beta\alpha}^{\gamma}, q_{\beta\alpha}^{\gamma}, r_{\beta\alpha}^{\gamma})$.

The skew-symmetric matrix of the angular rate vector is also commonly used:¹

$$\boldsymbol{\Omega}_{\beta\alpha}^{\gamma} = [\boldsymbol{\omega}_{\beta\alpha}^{\gamma} \wedge] = \begin{pmatrix} 0 & -\omega_{\beta\alpha 3}^{\gamma} & \omega_{\beta\alpha 2}^{\gamma} \\ \omega_{\beta\alpha 3}^{\gamma} & 0 & -\omega_{\beta\alpha 1}^{\gamma} \\ -\omega_{\beta\alpha 2}^{\gamma} & \omega_{\beta\alpha 1}^{\gamma} & 0 \end{pmatrix} \quad (2.26)$$

where both the rows and columns of $\boldsymbol{\Omega}_{\beta\alpha}^{\gamma}$ are resolved in the γ frame. Skew-symmetric matrices transform as

$$\boldsymbol{\Omega}_{\beta\alpha}^{\delta} = \mathbf{C}_{\gamma}^{\delta} \boldsymbol{\Omega}_{\beta\alpha}^{\gamma} \mathbf{C}_{\delta}^{\gamma} \quad (2.27)$$

where the left-hand coordinate transformation matrix transforms the rows of the skew-symmetric matrix and the other matrix transforms the columns. It can be shown, using the small angle approximation applied in the limit $\delta t \rightarrow 0$, that the time derivative of the coordinate transformation matrix is [6]:²

$$\dot{\mathbf{C}}_{\beta}^{\alpha} = -\mathbf{C}_{\beta}^{\alpha} \boldsymbol{\Omega}_{\beta\alpha}^{\beta} = -\boldsymbol{\Omega}_{\beta\alpha}^{\alpha} \mathbf{C}_{\beta}^{\alpha} = \mathbf{C}_{\beta}^{\alpha} \boldsymbol{\Omega}_{\alpha\beta}^{\beta} = \boldsymbol{\Omega}_{\alpha\beta}^{\alpha} \mathbf{C}_{\beta}^{\alpha} \quad (2.28)$$

2.2.6 Cartesian Position

As Figure 2.9 shows, the *Cartesian position* of the origin of frame α with respect to the origin of frame β , resolved about the axes of frame γ , is $\mathbf{r}_{\beta\alpha}^{\gamma} = (x_{\beta\alpha}^{\gamma}, y_{\beta\alpha}^{\gamma}, z_{\beta\alpha}^{\gamma})$, where x , y , and z are the components of position in the x , y , and z axes of the γ frame. Cartesian position differs from curvilinear position (Section 2.3.1) in that the resolving axes are independent of the position vector. It is also known as Euclidean position.

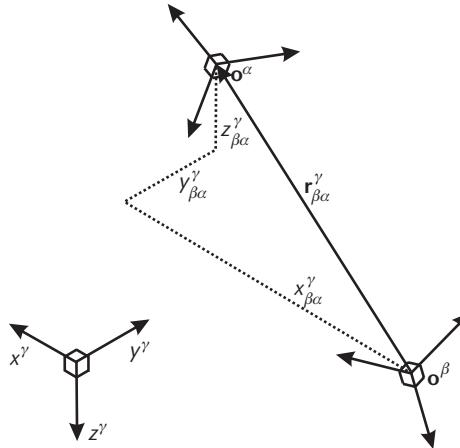


Figure 2.9 Position of the origin of frame α with respect to the origin of frame β in frame γ axes.

The object and reference frames of a Cartesian position may be transposed simply by reversing the sign:

$$\mathbf{r}_{\beta\alpha}^{\gamma} = -\mathbf{r}_{\alpha\beta}^{\gamma} \quad (2.29)$$

Similarly, two positions with common resolving axes may be added provided the object frame of one matches the reference frame of the other:

$$\mathbf{r}_{\beta\alpha}^{\gamma} = \mathbf{r}_{\beta\delta}^{\gamma} + \mathbf{r}_{\delta\alpha}^{\gamma} \quad (2.30)$$

Position may be resolved in a different frame by applying a coordinate transformation matrix:¹

$$\mathbf{r}_{\beta\alpha}^{\delta} = \mathbf{C}_{\gamma}^{\delta} \mathbf{r}_{\beta\alpha}^{\gamma} \quad (2.31)$$

Note that

$$\mathbf{r}_{\alpha\beta}^{\alpha} = -\mathbf{C}_{\beta}^{\alpha} \mathbf{r}_{\beta\alpha}^{\beta} \quad (2.32)$$

Considering specific frames, the origins of the ECI and ECEF frames coincide, as do those of the local navigation and body frames. Therefore

$$\mathbf{r}_{ie}^{\gamma} = \mathbf{r}_{nb}^{\gamma} = 0 \quad (2.33)$$

and

$$\mathbf{r}_{ib}^{\gamma} = \mathbf{r}_{eb}^{\gamma} = \mathbf{r}_{in}^{\gamma} = \mathbf{r}_{en}^{\gamma} \quad (2.34)$$

which also holds for the time derivatives.

2.2.7 Velocity

Velocity is defined as the rate of change of the position of the origin of an object frame with respect to the origin and axes of a reference frame. This may, in turn, be resolved about the axes of a third frame. Thus, the velocity of frame α with respect to frame β , resolved about the axes of frame γ , is³

$$\mathbf{v}_{\beta\alpha}^{\gamma} = \mathbf{C}_{\beta}^{\gamma} \dot{\mathbf{r}}_{\beta\alpha}^{\beta} \quad (2.35)$$

A velocity is thus registered if the object frame, α , moves with respect to the β -frame origin, or the reference frame, β , moves with respect to the α -frame origin. However, the velocity is defined not only with respect to the origin of the reference frame, but with respect to the axes as well. Therefore, a velocity is also registered if the reference frame, β , rotates with respect to the α -frame origin. This is important in navigation as many of the commonly used reference frames rotate with respect to each other. Figure 2.10 illustrates the three types of motion that register a velocity. No velocity is registered if the object frame rotates or the resolving frame, γ , moves or rotates with respect to the origin of the reference frame, β .

It should be noted that $\mathbf{v}_{\beta\alpha}^{\gamma}$ is not equal to the time derivative of $\mathbf{r}_{\beta\alpha}^{\beta}$ unless there is no angular motion of the resolving frame, γ , with respect to the reference frame, β . From (2.31) and (2.35),

$$\begin{aligned} \dot{\mathbf{r}}_{\beta\alpha}^{\gamma} &= \dot{\mathbf{C}}_{\beta}^{\gamma} \mathbf{r}_{\beta\alpha}^{\beta} + \mathbf{C}_{\beta}^{\gamma} \dot{\mathbf{r}}_{\beta\alpha}^{\beta} \\ &= \dot{\mathbf{C}}_{\beta}^{\gamma} \mathbf{r}_{\beta\alpha}^{\beta} + \mathbf{v}_{\beta\alpha}^{\gamma} \end{aligned} \quad (2.36)$$

Rotation between the resolving axes and the reference frame is important in navigation because the local navigation frame rotates with respect to the ECEF frame as the origin of the former moves with respect to the Earth.

Unlike with Cartesian position, the reference and object frames cannot be interchanged by reversing the sign unless there is no angular motion between them. The correct relationship is

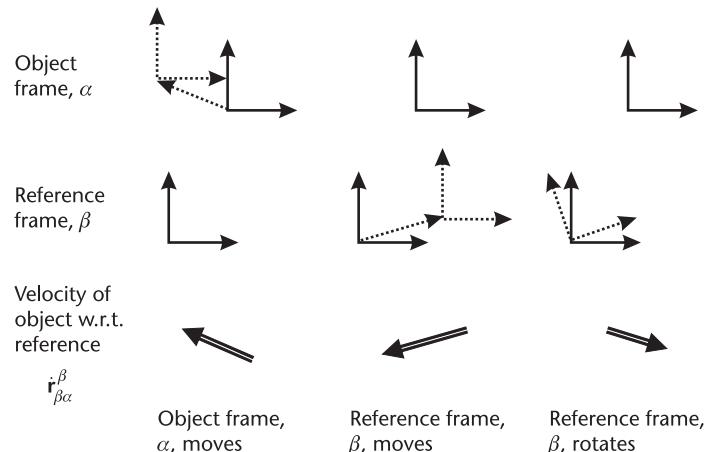


Figure 2.10 Motion causing a velocity to register.

$$\mathbf{v}_{\alpha\beta}^{\gamma} = -\mathbf{v}_{\beta\alpha}^{\gamma} - \mathbf{C}_{\alpha}^{\gamma} \dot{\mathbf{C}}_{\beta}^{\alpha} \mathbf{r}_{\beta\alpha}^{\beta} \quad (2.37)$$

although

$$\mathbf{v}_{\alpha\beta}^{\gamma} \Big|_{\dot{\mathbf{C}}_{\beta}^{\alpha} = 0} = -\mathbf{v}_{\beta\alpha}^{\gamma} \quad (2.38)$$

Similarly, velocities do not add if the reference frames rotate with respect to each other. Thus,

$$\mathbf{v}_{\beta\alpha}^{\gamma} \neq \mathbf{v}_{\beta\delta}^{\gamma} + \mathbf{v}_{\delta\alpha}^{\gamma} \quad (2.39)$$

although

$$\mathbf{v}_{\beta\alpha}^{\gamma} \Big|_{\dot{\mathbf{C}}_{\beta}^{\delta} = 0} = \mathbf{v}_{\beta\delta}^{\gamma} + \mathbf{v}_{\delta\alpha}^{\gamma} \quad (2.40)$$

Velocity may be transformed from one resolving frame to another using the appropriate coordinate transformation matrix:¹

$$\mathbf{v}_{\beta\alpha}^{\delta} = \mathbf{C}_{\gamma}^{\delta} \mathbf{v}_{\beta\alpha}^{\gamma} \quad (2.41)$$

As the ECI and ECEF frames have a common origin, as do the body and local navigation frames,

$$\mathbf{v}_{ie}^{\gamma} = \mathbf{v}_{nb}^{\gamma} = 0, \quad \mathbf{v}_{ib}^{\gamma} = \mathbf{v}_{in}^{\gamma}, \quad \mathbf{v}_{eb}^{\gamma} = \mathbf{v}_{en}^{\gamma} \quad (2.42)$$

However, because the ECEF frame rotates with respect to an inertial frame,

$$\mathbf{v}_{ib}^{\gamma} \neq \mathbf{v}_{eb}^{\gamma}, \quad \mathbf{v}_{in}^{\gamma} \neq \mathbf{v}_{en}^{\gamma} \quad (2.43)$$

even where the γ frame is the e or i frame.

The Earth-referenced velocity in local navigation frame axes, \mathbf{v}_{eb}^n or \mathbf{v}_{en}^n , is often abbreviated in the literature to \mathbf{v}^n and the inertial-referenced velocity, \mathbf{v}_{ib}^i or \mathbf{v}_{in}^i , to \mathbf{v}^i .²

Speed is simply the magnitude of the velocity, so $v_{\beta\alpha} = |\mathbf{v}_{\beta\alpha}|$.

2.2.8 Acceleration

Acceleration is defined as the second time derivative of the position of the origin of one frame with respect to the origin and axes of another frame. Thus, the acceleration of frame α with respect to frame β , resolved about the axes of frame γ , is³

$$\mathbf{a}_{\beta\alpha}^{\gamma} = \mathbf{C}_{\beta}^{\gamma} \ddot{\mathbf{r}}_{\beta\alpha}^{\beta} \quad (2.44)$$

The acceleration is the force per unit mass on the object applied from the reference frame. Its magnitude is necessarily independent of the resolving frame. It is not the same as the time derivative of $\mathbf{v}_{\beta\alpha}^\gamma$ or the second time derivative of $\mathbf{r}_{\beta\alpha}^\gamma$. These depend on the rotation of the resolving frame, γ , with respect to the reference frame, β :

$$\dot{\mathbf{v}}_{\beta\alpha}^\gamma = \dot{\mathbf{C}}_\beta^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\beta + \mathbf{a}_{\beta\alpha}^\gamma \quad (2.45)$$

$$\begin{aligned} \ddot{\mathbf{r}}_{\beta\alpha}^\gamma &= \ddot{\mathbf{C}}_\beta^\gamma \mathbf{r}_{\beta\alpha}^\beta + \dot{\mathbf{C}}_\beta^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\beta + \dot{\mathbf{v}}_{\beta\alpha}^\gamma \\ &= \ddot{\mathbf{C}}_\beta^\gamma \mathbf{r}_{\beta\alpha}^\beta + 2\dot{\mathbf{C}}_\beta^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\beta + \mathbf{a}_{\beta\alpha}^\gamma \end{aligned} \quad (2.46)$$

From (2.28) and (2.31),

$$\ddot{\mathbf{C}}_\beta^\gamma \mathbf{r}_{\beta\alpha}^\beta = (\boldsymbol{\Omega}_{\beta\gamma}^\gamma \boldsymbol{\Omega}_{\beta\gamma}^\gamma - \dot{\boldsymbol{\Omega}}_{\beta\gamma}^\gamma) \mathbf{r}_{\beta\alpha}^\gamma \quad (2.47)$$

while from (2.36), (2.28), and (2.31),

$$\begin{aligned} \dot{\mathbf{C}}_\beta^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\beta &= -\boldsymbol{\Omega}_{\beta\gamma}^\gamma \mathbf{C}_\beta^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\beta = \boldsymbol{\Omega}_{\beta\gamma}^\gamma (\dot{\mathbf{C}}_\beta^\gamma \mathbf{r}_{\beta\alpha}^\beta - \dot{\mathbf{r}}_{\beta\alpha}^\gamma) \\ &= -\boldsymbol{\Omega}_{\beta\gamma}^\gamma \boldsymbol{\Omega}_{\beta\gamma}^\gamma \mathbf{r}_{\beta\alpha}^\gamma - \boldsymbol{\Omega}_{\beta\gamma}^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\gamma \end{aligned} \quad (2.48)$$

Substituting these into (2.46) gives

$$\ddot{\mathbf{r}}_{\beta\alpha}^\gamma = -(\boldsymbol{\Omega}_{\beta\gamma}^\gamma \boldsymbol{\Omega}_{\beta\gamma}^\gamma + \dot{\boldsymbol{\Omega}}_{\beta\gamma}^\gamma) \mathbf{r}_{\beta\alpha}^\gamma - 2\boldsymbol{\Omega}_{\beta\gamma}^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\gamma + \mathbf{a}_{\beta\alpha}^\gamma \quad (2.49)$$

Thus, the motion of an object, expressed in the axes of a rotating resolving frame, such as the local navigation frame, depends not only on the applied force, but on two virtual forces, the centrifugal force and the Coriolis force [7]. The first term on the right-hand side of (2.46) is the centrifugal acceleration, and the second term, $-2\boldsymbol{\Omega}_{\beta\gamma}^\gamma \dot{\mathbf{r}}_{\beta\alpha}^\gamma$, is the Coriolis acceleration. The centripetal force is the real force that is applied to balance the centrifugal force in order to maintain a constant position vector expressed in rotating resolving axes.

Like velocities, accelerations do not add if the reference frames rotate with respect to each other:³

$$\mathbf{a}_{\beta\alpha}^\gamma \neq \mathbf{a}_{\beta\delta}^\gamma + \mathbf{a}_{\delta\alpha}^\gamma \quad (2.50)$$

but an acceleration may be resolved about a different set of axes by applying the appropriate coordinate transformation matrix:

$$\mathbf{a}_{\beta\alpha}^\delta = \mathbf{C}_\gamma^\delta \mathbf{a}_{\beta\alpha}^\gamma \quad (2.51)$$

2.3 Earth Surface and Gravity Models

For most applications, a position solution with respect to the Earth's surface is required. Obtaining this requires a reference surface to be defined with respect to

the center and axes of the Earth. A set of coordinates for expressing position with respect to that surface, the latitude, longitude, and height, must then be defined. To transform inertially referenced measurements to Earth referenced, the Earth's rotation must also be defined. This section addresses each of these issues in turn. It then explains the distinctions between specific force and acceleration and between gravity and gravitation, which are key concepts in inertial navigation.

2.3.1 The Ellipsoid Model of the Earth's Surface

The ECEF coordinate frame enables the user to navigate with respect to the center of the Earth. However, for most practical navigation problems, the user wants to know their position relative to the Earth's surface. The first step is to define that surface in the ECEF frame. Unfortunately, the Earth's surface is irregular. Modeling it accurately within a navigation system is not practical, requiring a large amount of data storage and more complex navigation algorithms. Therefore, the Earth's surface is approximated to a regular shape, which is then fitted to the true surface of the Earth at mean sea level.

The model of the Earth's surface used in most navigation systems is a type of ellipsoid, known as an oblate spheroid. Figure 2.11 depicts a cross-section of this reference ellipsoid, noting that this and subsequent diagrams exaggerate the flattening of the Earth. The ellipsoid exhibits rotational symmetry about the north-south (z^e) axis and mirror symmetry over the equatorial plane. It is defined by two radii. The equatorial radius, R_0 , or the length of the semi-major axis, a , is the distance from the center to any point on the equator, which is the furthest part of the surface from the center. The polar radius, R_p , or the length of the semi-minor axis, b , is the distance from the center to either pole, which are the nearest points on the surface to the center.

The ellipsoid is commonly defined in terms of the equatorial radius and either the (primary or major) eccentricity of the ellipsoid, e , or the flattening of the ellipsoid, f . These are defined by

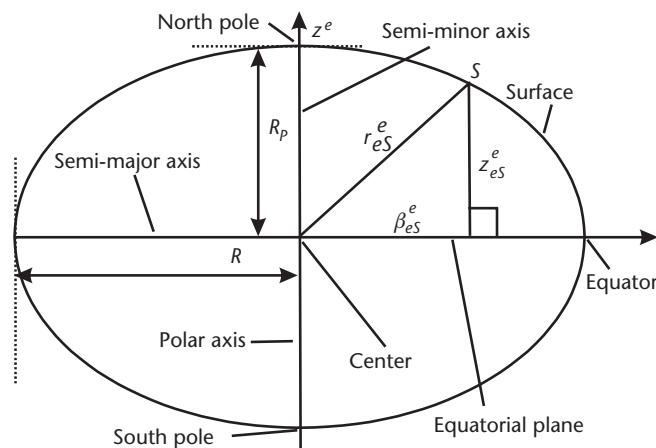


Figure 2.11 Cross-section of the ellipsoid representing the Earth's surface.

$$e = \sqrt{1 - \frac{R_p^2}{R_0^2}}, \quad f = \frac{R_0 - R_p}{R_0} \quad (2.52)$$

and are related by

$$e = \sqrt{2f - f^2}, \quad f = 1 - \sqrt{1 - e^2} \quad (2.53)$$

The Cartesian position of a point, S , on the ellipsoid surface is $\mathbf{r}_{eS}^e = (x_{eS}^e, y_{eS}^e, z_{eS}^e)$. The distance of that point from the center of the Earth is known as the geocentric radius and is simply

$$r_{eS}^e = |\mathbf{r}_{eS}^e| = \sqrt{(x_{eS}^e)^2 + (y_{eS}^e)^2 + (z_{eS}^e)^2} \quad (2.54)$$

It is useful to define the magnitude of the projection of \mathbf{r}_{eS}^e into the equatorial plane as β_{eS}^e . Thus,

$$\beta_{eS}^e = \sqrt{(x_{eS}^e)^2 + (y_{eS}^e)^2} \quad (2.55)$$

The cross-section of the ellipsoid shown in Figure 2.11 is the vertical plane containing the vector \mathbf{r}_{eS}^e . Thus z_{eS}^e and β_{eS}^e are constrained by the ellipse equation:

$$\left(\frac{\beta_{eS}^e}{R_0}\right)^2 + \left(\frac{z_{eS}^e}{R_p}\right)^2 = 1 \quad (2.56)$$

Substituting in (2.53) defines the surface of the ellipsoid:

$$\left(\frac{x_{eS}^e}{R_0}\right)^2 + \left(\frac{y_{eS}^e}{R_0}\right)^2 + \left(\frac{z_{eS}^e}{R_p}\right)^2 = 1 \quad (2.57)$$

As well as providing a reference for determining position, the ellipsoid model is also crucial in defining the local navigation frame (Section 2.1.3), as the down direction of this frame is defined as the normal to the ellipsoid, pointing to the equatorial plane.

Realizing the ellipsoid model in practice requires the positions of a large number of points on the Earth's surface to be measured. There is no practical method of measuring position with respect to the center of the Earth, noting that the center of an ellipsoid is not necessarily the center of mass. Consequently, position has been measured by surveying the relative positions of a number of points, a process known as triangulation. This has been done on national, regional, and continental bases, providing a host of different ellipsoid models, or datums, that provide a good fit to the Earth's surface across the area of interest, but a poor fit elsewhere in the world [8].

The advent of satellite navigation has enabled the position of points across the whole of the Earth's surface to be measured with respect to a common reference,

the satellite constellation, leading to the development of global ellipsoid models. The two main standards are the World Geodetic System 1984 (WGS 84) [9] and the international terrestrial reference frame (ITRF) [10]. Both of these datums have their origin at the Earth's center of mass and define rotation using the IRP/CTP.

WGS 84 was developed by the Defense Mapping Agency, now the National Geospatial-Intelligence Agency (NGA), as a standard for the U.S. military and is a refinement of predecessors WGS 60, WGS 66, and WGS 72. Its use for GPS and in most INSs led to its adoption as a global standard for navigation systems. WGS 84 was originally realized with 1691 Transit position fixes, each accurate to 1–2m and was revised in the 1990s using GPS measurements and ITRF data [11]. As well as defining an ECEF coordinate frame and an ellipsoid, WGS 84 provides models of the Earth's geoid (Section 2.3.3) and gravity field (Section 2.3.5) and a set of fundamental constants. WGS 84 defines the ellipsoid in terms of the equatorial radius and the flattening. The polar radius and eccentricity may be derived from this. The values are

- $R_0 = 6,378,137.0 \text{ m}$, $f = 1 / 298.257223563$
- $R_p = 6,356,752.3142 \text{ m}$, $e = 0.0818191908425$

The ITRF is maintained by the IERS and is the datum of choice for the scientific community, particularly geodesists. It is based on a mixture of measurements from satellite laser ranging, lunar laser ranging, very long baseline interferometry (VLBI), and GPS. ITRF is more precise than WGS 84, though the revision of the latter in the 1990s brought the two into closer alignment and WGS 84 is now considered to be a realization of the ITRF. Galileo will use a realization of the ITRF known as the Galileo terrestrial reference frame (GTRF), while GLONASS plans to switch to an ITRF-based datum from the PZ-90 datum. All datums must be regularly updated to account for plate tectonic motion, which causes the position of all points on the surface to move by a few centimeters each year with respect to the center of the Earth.

2.3.2 Curvilinear Position

Position with respect to the Earth's surface is described using three mutually orthogonal coordinates, aligned with the axes of the local navigation frame. The distance from the body described to the surface along the normal to that surface is the *height* or *altitude*. The north-south axis coordinate of the point on the surface where that normal intersects is the *latitude*, and the coordinate of that point in the east-west axis is the *longitude*. Each of these is defined in detail later. Because the orientation of all three axes with respect to the Earth varies with location, the latitude, longitude, and height are collectively known as *curvilinear* position.

Connecting all points on the ellipsoid surface of the same latitude produces a circle centered about the polar (north-south) axis; this is known as a *parallel* and has radius β_{eS}^e . Similarly, the points of constant longitude on the ellipsoid surface define a semi-ellipse, running from pole to pole, known as a *meridian*.

Traditionally, latitude was measured by determining the local vertical with a plumb bob and the axis of rotation from the motion of the stars. However, this

astronomical latitude has two drawbacks. First, due to local gravity variation, multiple points along a meridian can have the same astronomical latitude [8]. Second, as a result of polar motion, the latitude of all points varies slightly with time.

The *geocentric latitude*, Φ , illustrated in Figure 2.12, is the angle of intersection of the line from the center to a point on the surface of the ellipsoid with the equatorial plane. For all types of latitude, the convention is that latitude is positive in the northern hemisphere and negative in the southern hemisphere. By trigonometry, the geocentric latitude of a point S on the surface is given by

$$\tan \Phi_S = \frac{z_{eS}^e}{\beta_{eS}^e} = \frac{z_{eS}^e}{\sqrt{(x_{eS}^e)^2 + (y_{eS}^e)^2}}, \quad \sin \Phi_S = \frac{z_{eS}^e}{r_{eS}^e} \quad (2.58)$$

The *geodetic latitude*, L , also shown in Figure 2.12, is the angle of intersection of the normal to the ellipsoid with the equatorial plane. The symbol ϕ is also commonly used. Geodetic latitude is a rationalization of astronomical latitude, retaining the basic principle but removing the ambiguity. It is the standard form of latitude used in terrestrial navigation. As the geodetic latitude is defined by the normal to the surface, it can be obtained from the gradient of that surface. Thus, for a point S on the surface of the ellipsoid,

$$\tan L_S = -\frac{\partial \beta_{eS}^e}{\partial z_{eS}^e} \quad (2.59)$$

Differentiating (2.56) and then substituting (2.52) and (2.55),

$$\frac{\partial \beta_{eS}^e}{\partial z_{eS}^e} = -\frac{z_{eS}^e R_0^2}{\beta_{eS}^e R_P^2} = -\frac{z_{eS}^e}{(1 - e^2) \beta_{eS}^e} \quad (2.60)$$

Thus,

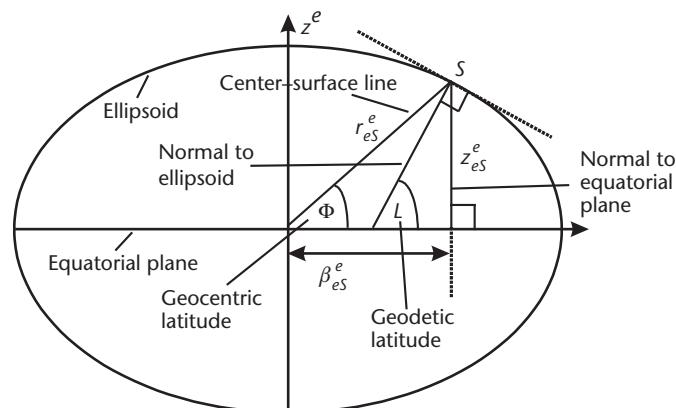


Figure 2.12 Geocentric and geodetic latitude.

$$\tan L_S = \frac{z_{eS}^e}{(1 - e^2) \beta_{eS}^e} = \frac{z_{eS}^e}{(1 - e^2) \sqrt{(x_{eS}^e)^2 + (y_{eS}^e)^2}} \quad (2.61)$$

Substituting in (2.58) and (2.59) gives the relationship between the geodetic and geocentric latitudes:

$$\tan \Phi_S = (1 - e^2) \tan L_S \quad (2.62)$$

For a body, b , which is not on the surface of the ellipsoid, the geodetic latitude is given by the coordinates of the point, $S(b)$, where the normal to the surface from that body intersects the surface. Thus,

$$\tan L_b = \frac{z_{eS(b)}^e}{(1 - e^2) \sqrt{(x_{eS(b)}^e)^2 + (y_{eS(b)}^e)^2}} \quad (2.63)$$

$$\tan L_b \neq \frac{z_{eb}^e}{(1 - e^2) \sqrt{(x_{eb}^e)^2 + (y_{eb}^e)^2}}$$

The *longitude*, λ , illustrated in Figure 2.13, is the angle subtended in the equatorial plane between the meridian plane containing the point of interest and the IRM/CZM. The IRM is defined as the mean value of the zero longitude determinations from the adopted longitudes of a number of observatories around the world. It is approximately, but not exactly, equal to the original British zero meridian at Greenwich, London. The convention is that longitude is positive for meridians to the east of the IRM, so longitudes are positive in the eastern hemisphere and negative in the western hemisphere. Alternatively, they may be expressed between 0° and 360° or 2π rad. Note that some authors use the symbol λ for latitude and either l , L , or ϕ for longitude. By trigonometry, the longitude of a point S on the surface and of any body, b , is given by

$$\tan \lambda_S = \frac{y_{eS}^e}{x_{eS}^e}, \quad \tan \lambda_b = \frac{y_{eb}^e}{x_{eb}^e} \quad (2.64)$$

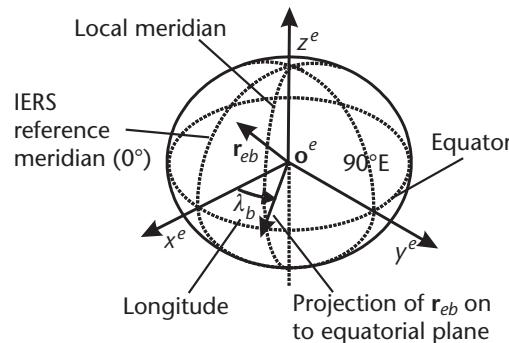


Figure 2.13 Illustration of longitude.

At this point, it is useful to define the radii of curvature of the ellipsoid. The radius of curvature for north-south motion is known as the meridian radius of curvature, R_N , and determines the rate of change of geodetic latitude along a meridian. It is the radius of curvature of a cross-section of the ellipsoid in the north-down plane at the point of interest. It varies with latitude and is smallest at the equator, where the geocentric radius is largest, and largest at the poles. It is given by

$$R_N(L) = \frac{R_0(1 - e^2)}{(1 - e^2 \sin^2 L)^{3/2}} \quad (2.65)$$

The radius of curvature for east-west motion is known as the transverse radius of curvature, R_E , and determines the rate of change of longitude along the surface normal to a meridian (which is not the same as a parallel). It is the radius of curvature of a cross-section of the ellipsoid in the east-down plane, not the plane of constant latitude, at the point of interest. It also varies with latitude and is smallest at the equator. The transverse radius of curvature is also equal to the length of the normal from a point on the surface to the polar axis. It is given by

$$R_E(L) = \frac{R_0}{\sqrt{1 - e^2 \sin^2 L}} \quad (2.66)$$

The transverse radius of curvature is also useful in defining the ellipsoid surface. From (2.56), (2.61), and (2.52), the radius of the circle of constant latitude and its distance from the equatorial plane are given by

$$\begin{aligned} \beta_{eS}^e &= R_E(L_S) \cos L_S \\ z_{eS}^e &= (1 - e^2) R_E(L_S) \sin L_S \end{aligned} \quad (2.67)$$

The *geodetic height*, h , sometimes known as the ellipsoidal height, is the distance from a body to the ellipsoid surface along the normal to that ellipsoid, with positive height denoting that the body is outside the ellipsoid. This is illustrated in Figure 2.14. By trigonometry, the height of a body, b , is given by

$$h_b = \frac{z_{eb}^e - z_{eS(b)}^e}{\sin L_b} \quad (2.68)$$

Substituting in (2.67),

$$h_b = \frac{z_{eb}^e}{\sin L_b} - (1 - e^2) R_E(L_b) \quad (2.69)$$

Using (2.55), (2.64), and (2.67–9), the Cartesian ECEF position may be obtained from the curvilinear position:

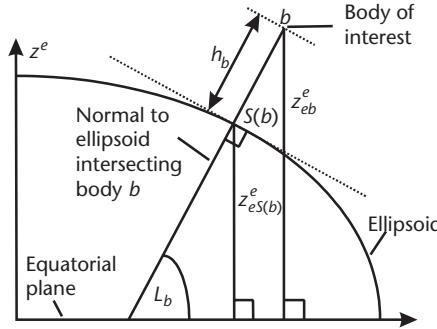


Figure 2.14 Height and geodetic latitude of a body, b .

$$\begin{aligned}x_{eb}^e &= (R_E(L_b) + h_b) \cos L_b \cos \lambda_b \\y_{eb}^e &= (R_E(L_b) + h_b) \cos L_b \sin \lambda_b \\z_{eb}^e &= [(1 - e^2) R_E(L_b) + h_b] \sin L_b\end{aligned}\quad (2.70)$$

The curvilinear position is obtained from the Cartesian ECEF position by implementing the inverse of the previous:

$$\begin{aligned}\sin L_b &= \frac{z_{eb}^e}{(1 - e^2) R_E(L_b) + h_b} \\ \tan \lambda_b &= \frac{y_{eb}^e}{x_{eb}^e} \\ h_b &= \frac{\sqrt{(x_{eb}^e)^2 + (y_{eb}^e)^2}}{\cos L_b} - R_E(L_b)\end{aligned}\quad (2.71)$$

where a four-quadrant arctangent function must be used for longitude. Note that, because R_E is a function of latitude, the latitude and height must be solved iteratively, taking $R_E = R_0$ on the first iteration. Great care should be taken when Cartesian and curvilinear position are mixed within a set of navigation equations to ensure that this computation is performed with sufficient precision. Otherwise, a divergent position solution could result. Alternatively, a closed-form solution is presented in some texts [4, 12].

The curvilinear position of a body, b , may be expressed in vector form as $\mathbf{p}_b = (L_b, \lambda_b, h_b)$. Note that only the object frame is specified as the ECEF reference frame, and local navigation frame resolving axes are implicit in the definition of curvilinear position.

The time derivative of curvilinear position is a linear function of the Earth-referenced velocity in local navigation frame axes:

$$\begin{aligned}\dot{L}_b &= \frac{v_{eb,N}^n}{R_N(L_b) + h_b} \\ \dot{\lambda}_b &= \frac{v_{eb,E}^n}{(R_E(L_b) + h_b) \cos L_b} \\ \dot{h}_b &= -v_{eb,D}^n\end{aligned}\tag{2.72}$$

This enables curvilinear position to be integrated directly from velocity without having to use Cartesian position as an intermediary.

Finally, although most navigation systems now use the WGS 84 datum, many maps are based on older datums. Consequently, it may be necessary to transform curvilinear position from one datum to another. The datums may use different origins, axis alignments, and scalings, as well as different radii curvature. Details of these transformations are outside the scope of a navigation systems book, so the reader is directed to geodesy texts (see the selected bibliography). No conversion between WGS 84 and ITRF position is needed, as the differences between the two datums are less than the uncertainty bounds.

2.3.3 The Geoid and Orthometric Height

The gravity potential is the potential energy required to overcome gravity (see Section 2.3.5). As water will always flow from an area of higher gravity potential to an area of lower gravity potential, mean sea level, which is averaged over the tide cycle, maintains a surface of equal gravity potential, or an equipotential surface. The geoid is a model of the Earth's surface that extends the mean sea level equipotential to include the land masses. Note that, over land, the physical surface of the Earth, known as the terrain, is generally above the geoid. As the Earth's gravity field varies with location, the geoid can differ from the ellipsoid by up to 100m. The height of the geoid with respect to the ellipsoid is denoted N . The current WGS 84 geoid model is known as the Earth Gravity Model 1996 (EGM 96) and has 130,676 ($= 360^2$) coefficients defining the geoid height, N , and gravitational potential as a spherical harmonic function of geodetic latitude and longitude [11]. The gravity vector at any point on the Earth's surface is perpendicular to the geoid, not the ellipsoid or the terrain, though, in practice, the difference is small.

The height of a body above the geoid is known as the *orthometric height* or the height above mean sea level and is denoted H . The orthometric height of the terrain is known as *elevation*. The orthometric height is related to the geodetic height by

$$H_b \approx h_b - N(L_b, \lambda_b)\tag{2.73}$$

This is not exact because the geodetic height is measured normal to the ellipsoid, whereas the orthometric height is measured normal to the geoid. Figure 2.15 illustrates the two heights, the geoid, ellipsoid, and terrain.

For many applications, orthometric height is more useful than geodetic height. Maps tend to express the height of the terrain and features with respect to the

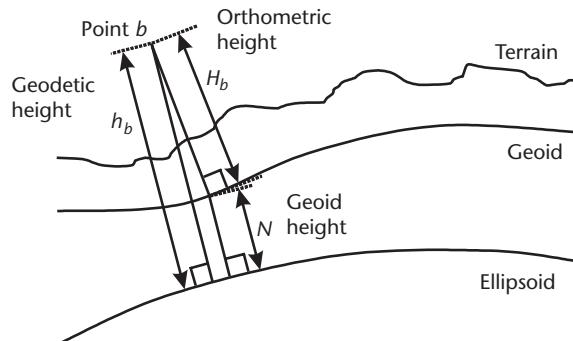


Figure 2.15 Height, geoid, ellipsoid, and terrain. (After: [13].)

geoid, making orthometric height critical for aircraft approach, landing, and low-level flight. It is also important in civil engineering, for example to determine the direction of flow of water. Thus, a navigation system will often need to incorporate a geoid model to convert between geodetic and orthometric height.

2.3.4 Earth Rotation

The Earth rotates, with respect to space, clockwise about the common z -axis of the ECI and ECEF frames. This is shown in Figure 2.16. Thus the Earth-rotation vector resolved in these axes is given by

$$\boldsymbol{\omega}_{ie}^i = \boldsymbol{\omega}_{ie}^e = \begin{pmatrix} 0 \\ 0 \\ \omega_{ie} \end{pmatrix} \quad (2.74)$$

The Earth-rotation vector resolved into local navigation frame axes is a function of geodetic latitude:

$$\boldsymbol{\omega}_{ie}^n = \begin{pmatrix} \omega_{ie} \cos L_b \\ 0 \\ -\omega_{ie} \sin L_b \end{pmatrix} \quad (2.75)$$

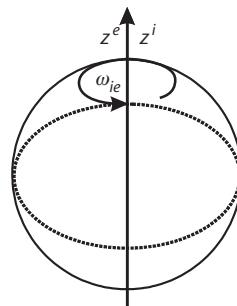


Figure 2.16 Earth rotation in the ECI and ECEF frames. (From: [4]. © 2002 QinetiQ Ltd. Reprinted with permission.)

The period of rotation of the Earth with respect to space is known as the sidereal day and is about 23 hours, 56 minutes, 4 seconds. This differs from the 24-hour mean solar day as the Earth's orbital motion causes the Earth-Sun direction with respect to space to vary, resulting in one more rotation than solar day each year (note that 1/365 of a day is about 4 minutes). The rate of rotation is not constant and the sidereal day can vary by several milliseconds from day to day. There are random changes due to wind and seasonal changes as ice forming and melting alters the Earth's moment of inertia. There is also a long-term reduction of the Earth rotation rate due to tidal friction [13].

For navigation purposes, a constant rotation rate is assumed, based on the mean sidereal day. The WGS 84 value of the Earth's angular rate is $\omega_{ie} = 7.292115 \times 10^{-5} \text{ rad s}^{-1}$ [9].

2.3.5 Specific Force, Gravitation, and Gravity

Specific force is the nongravitational force per unit mass on a body, sensed with respect to an inertial frame. It has no meaning with respect to any other frame, though it can be expressed in any axes. *Gravitation* is the fundamental mass attraction force; it does not incorporate any centripetal components.¹

Specific force is what people and instruments sense. Gravitation is not sensed because it acts equally on all points, causing them to move together. Other forces are sensed as they are transmitted from point to point. The sensation of weight is caused by the forces opposing gravity. There is no sensation of weight during freefall, where the specific force is zero. Conversely, under zero acceleration, the reaction to gravitation is sensed, and the specific force is equal and opposite to the acceleration due to gravitation. Figure 2.17 illustrates this for a mass on a spring. In both cases, the gravitational force on the mass is the same. However, in the stationary case, the spring exerts an opposite force.

A further example is provided by the upward motion of an elevator, illustrated in Figure 2.18. As the elevator accelerates upward, the specific force is higher and the occupants appear to weigh more. As the elevator decelerates, the specific force is lower than normal and the occupants feel lighter. In a windowless elevator, this

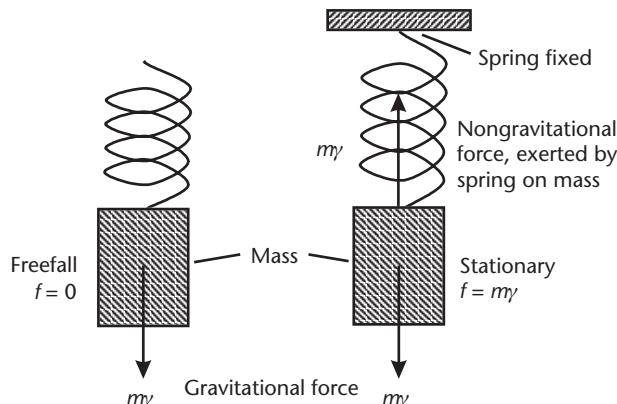


Figure 2.17 Forces on a mass on a spring.

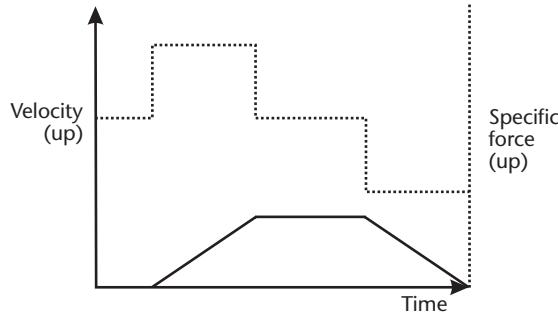


Figure 2.18 Velocity and specific force of an elevator moving up. (From: [4]. © 2002 QinetiQ Ltd. Reprinted with permission.)

can create the illusion that the elevator has overshot the destination floor and is dropping down to correct for it.

Thus, specific force, f , varies with acceleration, a , and the acceleration due to the gravitational force, γ , as

$$\mathbf{f}_{ib}^\gamma = \mathbf{a}_{ib}^\gamma - \boldsymbol{\gamma}_{ib}^\gamma \quad (2.76)$$

Specific force is the quantity measured by accelerometers. The measurements are made in the body frame of the accelerometer triad; thus, the sensed specific force is \mathbf{f}_{ib}^b .

Before defining gravity, it is useful to consider an object that is stationary with respect to a rotating frame, such as the ECEF frame. This has the properties

$$\mathbf{v}_{eb}^e = 0 \quad \mathbf{a}_{eb}^e = 0 \quad (2.77)$$

From (2.35) and (2.44), and applying (2.34),

$$\dot{\mathbf{r}}_{ib}^e = \dot{\mathbf{r}}_{eb}^e = 0 \quad \ddot{\mathbf{r}}_{ib}^e = \ddot{\mathbf{r}}_{eb}^e = 0 \quad (2.78)$$

The inertially referenced acceleration in ECEF frame axes is given by (2.49), noting that $\dot{\boldsymbol{\Omega}}_{ie}^e = 0$ as the Earth rate is assumed constant:

$$\mathbf{a}_{ib}^e = \boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{ib}^e + 2\boldsymbol{\Omega}_{ie}^e \dot{\mathbf{r}}_{ib}^e + \ddot{\mathbf{r}}_{ib}^e \quad (2.79)$$

Applying (2.78),

$$\mathbf{a}_{ib}^e = \boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e \quad (2.80)$$

Substituting this into the specific force definition, (2.76), gives

$$\mathbf{f}_{ib}^e = \boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e - \boldsymbol{\gamma}_{ib}^e \quad (2.81)$$

The specific force sensed when stationary with respect to the Earth frame is the reaction to what is known as the acceleration due to *gravity*, which is thus defined by²

$$\mathbf{g}_b^\gamma = -\mathbf{f}_{ib}^\gamma \Big|_{\mathbf{a}_{eb}^\gamma = 0, \mathbf{v}_{eb}^\gamma = 0} \quad (2.82)$$

Therefore, from (2.81), the acceleration due to gravity is

$$\mathbf{g}_b^\gamma = \boldsymbol{\gamma}_{ib}^\gamma - \boldsymbol{\Omega}_{ie}^\gamma \boldsymbol{\Omega}_{ie}^\gamma \mathbf{r}_{eb}^\gamma \quad (2.83)$$

noting from (2.74) and (2.75) that

$$\begin{aligned} \mathbf{g}_b^e &= \boldsymbol{\gamma}_{ib}^e + \omega_{ie}^2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{r}_{eb}^e \\ \mathbf{g}_b^n &= \boldsymbol{\gamma}_{ib}^n + \omega_{ie}^2 \begin{pmatrix} \sin^2 L_b & 0 & \cos L_b \sin L_b \\ 0 & 1 & 0 \\ \cos L_b \sin L_b & 0 & \cos^2 L_b \end{pmatrix} \mathbf{r}_{eb}^n \end{aligned} \quad (2.84)$$

The first term in (2.83) and (2.84) is the gravitational acceleration. The second term is the outward centrifugal acceleration due to the Earth's rotation, noting that this is a virtual force arising from the use of rotating resolving axes (see Section 2.2.8). From the inertial frame perspective, a centripetal acceleration, (2.80), is applied to maintain an object stationary with respect to a rotating frame. It is important not to confuse gravity, \mathbf{g} , with gravitation, $\boldsymbol{\gamma}$. At the Earth's surface, the total acceleration due to gravity is about 9.8 m s^{-2} , with the centrifugal component contributing up to 0.034 m s^{-2} . In orbit, the gravitational component is smaller and the centrifugal component larger.

The centrifugal component of gravity can be calculated exactly at all locations, but calculation of the gravitational component is more complex. For air applications, it is standard practice to use an empirical model of the surface gravity, g_0 , and apply a simple scaling law to calculate the variation with height.³

The WGS 84 datum [9] provides a simple model of the acceleration due to gravity at the ellipsoid as a function of latitude:

$$g_0(L) \approx 9.7803253359 \frac{(1 + 0.001931853 \sin^2 L)}{\sqrt{1 - e^2 \sin^2 L}} \text{ m s}^{-2} \quad (2.85)$$

This is known as the Somigliana model. Note that it is a gravity field model, not a gravitational field model. The geoid (Section 2.3.3) defines a surface of constant gravity potential. However, the acceleration due to gravity is obtained from the gradient of the gravity potential, so is not constant across the geoid. Although the true gravity vector is perpendicular to the geoid (not the terrain),

it is a reasonable approximation for most navigation applications to treat it as perpendicular to the ellipsoid. Thus,

$$\mathbf{g}_0^\gamma(L) \approx g_0(L) \mathbf{u}_{nD}^\gamma \quad (2.86)$$

where \mathbf{u}_{nD}^γ is the down unit vector of the local navigation frame.

The gravitational acceleration at the ellipsoid can be obtained from the acceleration due to gravity by subtracting the centrifugal acceleration. Thus,

$$\boldsymbol{\gamma}_0^\gamma(L) = \mathbf{g}_0^\gamma(L) + \boldsymbol{\Omega}_{ie}^\gamma \boldsymbol{\Omega}_{ie}^\gamma \mathbf{r}_{eS}^\gamma(L) \quad (2.87)$$

where, from (2.65), the geocentric radius at the surface is given by

$$\mathbf{r}_{eS}^e(L) = R_E(L) \sqrt{\cos^2 L + (1 - e^2)^2 \sin^2 L} \quad (2.88)$$

The gravitational field varies roughly as that for a point mass, so gravitational acceleration can be scaled with height as

$$\boldsymbol{\gamma}_{ib}^\gamma \approx \frac{(r_{eS}^e(L_b))^2}{(r_{eS}^e(L_b) + h_b)^2} \boldsymbol{\gamma}_0^\gamma(L_b) \quad (2.89)$$

For heights less than about 10 km, the scaling can be further approximated to $(1 - 2h_b/r_{eS}^e(L_b))$. The acceleration due to gravity, \mathbf{g} , may then be recombined using (2.83). As the centrifugal component of gravity is small, it is reasonable to apply the height scaling to \mathbf{g} where the height is small and/or poor quality accelerometers are used. Alternatively, a more accurate set of formulae for calculating gravity as a function of latitude and height is given in [9]. An approximation for the variation of the down component with height is

$$g_{b,D}^n(L_b, h_b) \approx g_0(L_b) \left[1 - \frac{2}{R_0} \left(1 + f + \frac{\omega_{ie}^2 R_0^2 R_p}{\mu} \right) h_b + \frac{3}{R_0^2} h_b^2 \right] \quad (2.90)$$

where μ is the Earth's gravitational constant and its WGS 84 value [9] is $3.986004418 \times 10^{14} \text{ m}^3 \text{s}^{-2}$.

When working in an inertial reference frame, only the gravitational acceleration is required. This can be calculated directly at varying height using [14]

$$\boldsymbol{\gamma}_{ib}^i = -\frac{\mu}{|\mathbf{r}_{ib}^i|^3} \left\{ \mathbf{r}_{ib}^i + \frac{3}{2} J_2 \frac{R_0^2}{|\mathbf{r}_{ib}^i|^2} \left\{ \begin{aligned} & \left[1 - 5(r_{ib,z}^i / |\mathbf{r}_{ib}^i|)^2 \right] r_{ib,x}^i \\ & \left[1 - 5(r_{ib,z}^i / |\mathbf{r}_{ib}^i|)^2 \right] r_{ib,y}^i \\ & \left[3 - 5(r_{ib,z}^i / |\mathbf{r}_{ib}^i|)^2 \right] r_{ib,z}^i \end{aligned} \right\} \right\} \quad (2.91)$$

where J_2 is the Earth's second gravitational constant and takes the value 1.082627×10^{-3} [9].

Much higher precision may be obtained using a spherical harmonic model, such as the 360^2 coefficient EGM 96 gravity model [11]. Further precision is given by a gravity anomaly database, which comprises the difference between the measured and modeled gravity fields over a grid of locations. Gravity anomalies tend to be largest over major mountain ranges and ocean trenches.

2.4 Frame Transformations

An essential feature of navigation mathematics is the capability to transform kinematics between coordinate frames. This section summarizes the equations for expressing the attitude of one frame with respect to another and transforming Cartesian position, velocity, acceleration, and angular rate between references to the inertial, Earth, and local navigation frames. The section concludes with the equations for transposing a navigation solution from one object to another.¹

Cartesian position, velocity, acceleration, and angular rate referenced to the same frame transform between resolving axes simply by applying the coordinate transformation matrix (2.7):

$$\mathbf{x}_{\beta\alpha}^\gamma = \mathbf{C}_\delta^\gamma \mathbf{x}_{\beta\alpha}^\delta \quad \mathbf{x} \in \mathbf{r}, \mathbf{v}, \mathbf{a}, \boldsymbol{\omega} \quad \gamma, \delta \in i, e, n, b \quad (2.92)$$

Therefore, these transforms are not presented explicitly for each pair of frames.² The coordinate transformation matrices involving the body frame—that is,

$$\mathbf{C}_b^\beta, \mathbf{C}_\beta^b \quad \beta \in i, e, n$$

—describe the attitude of that body with respect to a reference frame. The body attitude with respect to a new reference frame may be obtained simply by multiplying by the coordinate transformation matrix between the two reference frames:

$$\mathbf{C}_b^\delta = \mathbf{C}_\beta^\delta \mathbf{C}_b^\beta \quad \mathbf{C}_\delta^b = \mathbf{C}_\beta^b \mathbf{C}_\delta^\beta \quad \beta, \delta \in i, e, n \quad (2.93)$$

Transforming Euler, quaternion, or rotation vector attitude to a new reference frame is more complex. One solution is to convert to the coordinate transformation matrix representation, transform the reference, and then convert back.

2.4.1 Inertial and Earth Frames

The center and z -axes of the ECI and ECEF coordinate frames are coincident. The x - and y -axes are coincident at time t_0 , and the frames rotate about the z axes at ω_{ie} (see Section 2.3.4). Thus,¹

$$\mathbf{C}_i^e = \begin{pmatrix} \cos \omega_{ie} (t - t_0) & \sin \omega_{ie} (t - t_0) & 0 \\ -\sin \omega_{ie} (t - t_0) & \cos \omega_{ie} (t - t_0) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (2.94)$$

$$\mathbf{C}_e^i = \begin{pmatrix} \cos \omega_{ie} (t - t_0) & -\sin \omega_{ie} (t - t_0) & 0 \\ \sin \omega_{ie} (t - t_0) & \cos \omega_{ie} (t - t_0) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Positions referenced to the two frames are the same, so only the resolving axes need be transformed:

$$\mathbf{r}_{eb}^e = \mathbf{C}_i^e \mathbf{r}_{ib}^i, \quad \mathbf{r}_{ib}^i = \mathbf{C}_e^i \mathbf{r}_{eb}^e \quad (2.95)$$

Velocity and acceleration transformation is more complex:

$$\begin{aligned} \mathbf{v}_{eb}^e &= \mathbf{C}_i^e (\mathbf{v}_{ib}^i - \boldsymbol{\Omega}_{ie}^i \mathbf{r}_{ib}^i) \\ \mathbf{v}_{ib}^i &= \mathbf{C}_e^i (\mathbf{v}_{eb}^e + \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e) \end{aligned} \quad (2.96)$$

$$\begin{aligned} \mathbf{a}_{eb}^e &= \mathbf{C}_i^e (\mathbf{a}_{ib}^i - 2\boldsymbol{\Omega}_{ie}^i \mathbf{v}_{ib}^i - \boldsymbol{\Omega}_{ie}^i \boldsymbol{\Omega}_{ie}^i \mathbf{r}_{ib}^i) \\ \mathbf{a}_{ib}^i &= \mathbf{C}_e^i (\mathbf{a}_{eb}^e + 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e + \boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e) \end{aligned} \quad (2.97)$$

Angular rates transform as

$$\boldsymbol{\omega}_{eb}^e = \mathbf{C}_i^e \left(\boldsymbol{\omega}_{ib}^i - \begin{pmatrix} 0 \\ 0 \\ \omega_{ie} \end{pmatrix} \right), \quad \boldsymbol{\omega}_{ib}^i = \mathbf{C}_e^i \left(\boldsymbol{\omega}_{eb}^e + \begin{pmatrix} 0 \\ 0 \\ \omega_{ie} \end{pmatrix} \right) \quad (2.98)$$

2.4.2 Earth and Local Navigation Frames

The relative orientation of the Earth and local navigation frames is determined by the geodetic latitude, L_b , and longitude, λ_b , of the body frame whose center coincides with that of the local navigation frame:¹

$$\mathbf{C}_e^n = \begin{pmatrix} -\sin L_b \cos \lambda_b & -\sin L_b \sin \lambda_b & \cos L_b \\ -\sin \lambda_b & \cos \lambda_b & 0 \\ -\cos L_b \cos \lambda_b & -\cos L_b \sin \lambda_b & -\sin L_b \end{pmatrix} \quad (2.99)$$

$$\mathbf{C}_n^e = \begin{pmatrix} -\sin L_b \cos \lambda_b & -\sin \lambda_b & -\cos L_b \cos \lambda_b \\ -\sin L_b \sin \lambda_b & \cos \lambda_b & -\cos L_b \sin \lambda_b \\ \cos L_b & 0 & -\sin L_b \end{pmatrix}$$

Position, velocity, and acceleration referenced to the local navigation frame are meaningless as the body frame center coincides with the navigation frame center. The resolving axes of Earth-referenced position, velocity, and acceleration are transformed using (2.92).

Angular rates transform as

$$\begin{aligned}\boldsymbol{\omega}_{nb}^n &= \mathbf{C}_e^n(\boldsymbol{\omega}_{eb}^e - \boldsymbol{\omega}_{en}^e), \quad \boldsymbol{\omega}_{eb}^e = \mathbf{C}_n^e(\boldsymbol{\omega}_{nb}^n + \boldsymbol{\omega}_{en}^n), \\ &= \mathbf{C}_e^n \boldsymbol{\omega}_{eb}^e - \boldsymbol{\omega}_{en}^n\end{aligned}\quad (2.100)$$

noting that a solution for $\boldsymbol{\omega}_{en}^n$ is obtained in Section 5.3.1.

2.4.3 Inertial and Local Navigation Frames

The inertial-local navigation frame coordinate transforms are obtained by multiplying (2.94) and (2.99)¹:

$$\begin{aligned}\mathbf{C}_i^n &= \begin{pmatrix} -\sin L_b \cos(\lambda_b + \omega_{ie}(t - t_0)) & -\sin L_b \sin(\lambda_b + \omega_{ie}(t - t_0)) & \cos L_b \\ -\sin(\lambda_b + \omega_{ie}(t - t_0)) & \cos(\lambda_b + \omega_{ie}(t - t_0)) & 0 \\ -\cos L_b \cos(\lambda_b + \omega_{ie}(t - t_0)) & -\cos L_b \sin(\lambda_b + \omega_{ie}(t - t_0)) & -\sin L_b \end{pmatrix} \\ \mathbf{C}_n^i &= \begin{pmatrix} -\sin L_b \cos(\lambda_b + \omega_{ie}(t - t_0)) & -\sin(\lambda_b + \omega_{ie}(t - t_0)) & -\cos L_b \cos(\lambda_b + \omega_{ie}(t - t_0)) \\ -\sin L_b \sin(\lambda_b + \omega_{ie}(t - t_0)) & \cos(\lambda_b + \omega_{ie}(t - t_0)) & -\cos L_b \sin(\lambda_b + \omega_{ie}(t - t_0)) \\ \cos L_b & 0 & -\sin L_b \end{pmatrix}\end{aligned}\quad (2.101)$$

Earth-referenced velocity and acceleration in navigation frame axes transform to and from their inertial frame inertial reference counterparts as

$$\begin{aligned}\mathbf{v}_{eb}^n &= \mathbf{C}_i^n(\mathbf{v}_{ib}^i - \boldsymbol{\Omega}_{ib}^i \mathbf{r}_{ib}^i) \\ \mathbf{v}_{ib}^i &= \mathbf{C}_n^i \mathbf{v}_{eb}^n + \mathbf{C}_e^i \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e\end{aligned}\quad (2.102)$$

$$\begin{aligned}\mathbf{a}_{eb}^n &= \mathbf{C}_i^n(\mathbf{a}_{ib}^i - 2\boldsymbol{\Omega}_{ie}^i \mathbf{v}_{ib}^i - \boldsymbol{\Omega}_{ie}^i \boldsymbol{\Omega}_{ie}^i \mathbf{r}_{ib}^i) \\ \mathbf{a}_{ib}^i &= \mathbf{C}_n^i(\mathbf{a}_{eb}^n + 2\boldsymbol{\Omega}_{ie}^n \mathbf{v}_{eb}^n) + \mathbf{C}_e^i \boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e\end{aligned}\quad (2.103)$$

Angular rates transform as

$$\begin{aligned}\boldsymbol{\omega}_{nb}^n &= \mathbf{C}_i^n(\boldsymbol{\omega}_{ib}^i - \boldsymbol{\omega}_{in}^i) \\ &= \mathbf{C}_i^n(\boldsymbol{\omega}_{ib}^i - \boldsymbol{\omega}_{ie}^i) - \boldsymbol{\omega}_{en}^n \\ \boldsymbol{\omega}_{ib}^i &= \mathbf{C}_n^i(\boldsymbol{\omega}_{nb}^n + \boldsymbol{\omega}_{in}^n) \\ &= \mathbf{C}_n^i(\boldsymbol{\omega}_{nb}^n + \boldsymbol{\omega}_{en}^n) + \boldsymbol{\omega}_{ie}^i\end{aligned}\quad (2.104)$$

2.4.4 Transposition of Navigation Solutions

Sometimes, there is a requirement to transpose a navigation solution from one position to another on a vehicle, such as between an INS and a GPS antenna, between an INS and the center of gravity, or between a reference and an aligning INS. Here, the equations for transposing position, velocity, and attitude from describing the b frame to describing the B frame are presented.¹

Let the orientation of frame B with respect to frame b be C_B^B and the position of frame B with respect to frame b in frame b axes be \mathbf{l}_{bb}^b , which is known as the *lever arm* or *moment arm*. Note that the lever arm is mathematically identical to the Cartesian position with B as the object frame and b as the reference and resolving frames. Figure 2.19 illustrates this.

Attitude transformation is straightforward:

$$\begin{aligned}\mathbf{C}_\beta^B &= \mathbf{C}_b^B \mathbf{C}_\beta^b \\ \mathbf{C}_B^\beta &= \mathbf{C}_b^\beta \mathbf{C}_B^b\end{aligned}\quad (2.105)$$

Cartesian position may be transposed using

$$\mathbf{r}_{\beta B}^\gamma = \mathbf{r}_{\beta b}^\gamma + \mathbf{C}_b^\gamma \mathbf{l}_{bb}^b \quad (2.106)$$

Precise transformation of latitude, longitude, and height requires conversion to Cartesian position and back. However, if the small angle approximation is applied to $1/R$, where R is the Earth radius, a simpler form may be used:

$$\begin{pmatrix} L_B \\ \lambda_B \\ h_B \end{pmatrix} \approx \begin{pmatrix} L_b \\ \lambda_b \\ h_b \end{pmatrix} + \begin{pmatrix} 1/(R_N(L_b) + h_b) & 0 & 0 \\ 0 & 1/[(R_E(L_b) + h_b) \cos L_b] & 0 \\ 0 & 0 & -1 \end{pmatrix} \mathbf{C}_b^n \mathbf{l}_{bb}^b \quad (2.107)$$

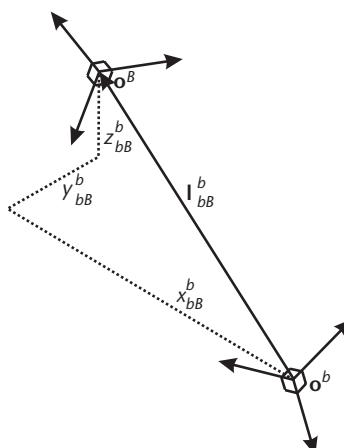


Figure 2.19 The lever arm from frame b to frame B .

Finally, the velocity transposition is obtained by differentiating (2.106) and substituting it into (2.35):

$$\mathbf{v}_{\beta B}^{\gamma} = \mathbf{v}_{\beta b}^{\gamma} + \mathbf{C}_{\beta}^{\gamma} \dot{\mathbf{C}}_{\beta}^{\beta} \mathbf{l}_{bb}^b \quad (2.108)$$

assuming \mathbf{l}_{bb}^b is constant. Substituting (2.28),²

$$\mathbf{v}_{\beta B}^{\gamma} = \mathbf{v}_{\beta b}^{\gamma} + \mathbf{C}_{\beta}^{\gamma} (\boldsymbol{\omega}_{\beta b}^b \wedge \mathbf{l}_{bb}^b) \quad (2.109)$$

References

- [1] Groves, P. D., Principles of Integrated Navigation (course notes), QinetiQ Ltd., 2002.
- [2] Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, New York: Wiley, 2001.
- [3] Titterton, D. H., and J. L. Weston, *Strapdown Inertial Navigation Technology*, Second Edition, Stevenage, United Kingdom: IEE, 2004.
- [4] Farrell, J. A., and M. Barth, *The Global Positioning System and Inertial Navigation*, New York: McGraw-Hill, 1999.
- [5] Bortz, J. E., “A New Mathematical Formulation for Strapdown Inertial Navigation,” *IEEE Trans. on Aerospace and Electronic Systems*, Vol. AES-7, No. 1, 1971, pp. 61–66.
- [6] Rogers, R. M., *Applied Mathematics in Integrated Navigation Systems*, Reston, VA: AIAA, 2000.
- [7] Feynman, R. P., R. B. Leighton, and M. Sands, *The Feynman Lectures on Physics, Volume 1*, Reading, MA: Addison-Wesley, 1963.
- [8] Ashkenazi, V., “Coordinate Systems: How to Get Your Position Very Precise and Completely Wrong,” *Journal of Navigation*, Vol. 39, No. 2, 1986, pp. 269–278.
- [9] Anon., *Department of Defense World Geodetic System 1984*, National Imagery and Mapping Agency (now NGA), TR8350.2, 3rd ed., 1997.
- [10] Boucher, C., et al., *The ITRF 2000*, International Earth Rotation and Reference Systems Service Technical Note No. 31, 2004.
- [11] Malys, S., et al., “Refinements to the World Geodetic System, 1984,” *Proc. ION GPS-97*, Kansas, MO, September 1997, pp. 915–920.
- [12] Kaplan, E. D., et al., “Fundamentals of Satellite Navigation,” in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan, and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 21–65.
- [13] Misra, P., and P. Enge, *Global Positioning System Signals, Measurements, and Performance*, Lincoln, MA: Ganga-Jamuna Press, 2001.
- [14] Britting, K. R., *Inertial Navigation Systems Analysis*, New York: Wiley, 1971.

Selected Bibliography

- Bomford, G., *Geodesy*, 4th ed., London, U.K.: Clarendon Press, 1980.
- Smith, J. R., *Introduction to Geodesy: The History and Concepts of Modern Geodesy*, New York: Wiley, 1997.
- Torge, W., *Geodesy*, Berlin, Germany: de Gruyter, 2001.

Endnotes

1. This and subsequent paragraphs are based on material written by the author for QinetiQ, so comprise QinetiQ copyright material.
2. End of QinetiQ copyright material.
3. This paragraph, up to this point, is based on material written by the author for QinetiQ, so comprises QinetiQ copyright material.

The Kalman Filter

The Kalman filter (KF) forms the basis of the vast majority of estimation algorithms used in navigation systems. Its uses include maintaining a smoothed satellite navigation solution, alignment and calibration of an INS, and integration of an INS with GNSS user environment and other navigation sensors. It is key to obtaining an optimal navigation solution from the various measurements available in a navigation system.

This chapter provides an introduction to the Kalman filter and a review of how it may be adapted for practical use in navigation applications. Section 3.1 provides a qualitative description of the Kalman filter, with the algorithm and mathematical models introduced in Section 3.2. Section 3.3 discusses the practical application of the Kalman filter, while Section 3.4 reviews some more advanced estimation techniques, based on the Kalman filter, that are relevant to navigation problems. This includes the Kalman smoother, which can give improved performance in post-processed applications. Examples of the Kalman filter's applications in navigation are presented within Chapters 7 and 12 through 14. For a more formalized and detailed treatment of Kalman filters, there are many applied mathematics books devoted solely to this subject [1–5].

3.1 Introduction

The Kalman filter is an estimation algorithm, rather than a filter. The basic technique was invented by R. E. Kalman in 1960 [6] and has been developed further by numerous authors since. It maintains real-time estimates of a number of parameters of a system, such as its position and velocity, that may continually change. The estimates are updated using a stream of measurements that are subject to noise. The measurements must be functions of the parameters estimated, but the set of measurements at a given time need not contain sufficient information to uniquely determine the values of the parameters at the time.

The Kalman filter uses knowledge of the deterministic and statistical properties of the system parameters and the measurements to obtain optimal estimates given the information available. To do this, it must carry more information from iteration to iteration than just the parameter estimates. Therefore, the Kalman filter also maintains a set of uncertainties in its estimates and a measure of the correlations between the errors in the estimates of the different parameters.

The Kalman filter is a Bayesian estimation technique. It is supplied with an initial set of estimates and then operates recursively, updating its working estimates

as a weighted average of their previous values and new values derived from the latest measurement data. By contrast, nonrecursive estimation algorithms derive their parameter estimates from the whole set of measurement data without prior estimates. For real-time applications, such as navigation, the recursive approach is more processor efficient, as only the new measurement data need be processed on each iteration. Old measurement data may be discarded.

This section provides a qualitative description of the Kalman filter and the steps forming its algorithm. Some brief examples of Kalman filter applications conclude the section. A quantitative description and derivation follow in Section 3.2.

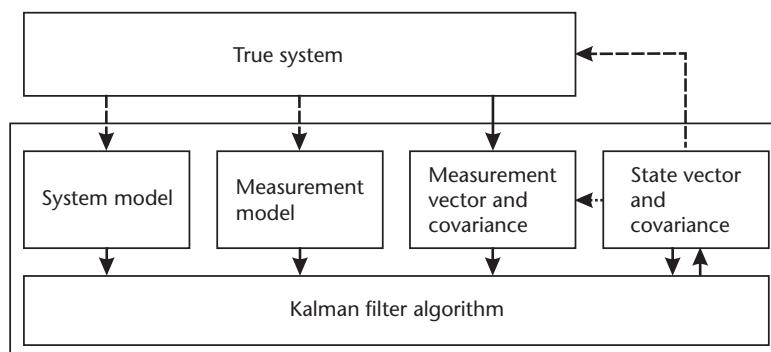
3.1.1 Elements and Phases of the Kalman Filter

Figure 3.1 shows the five core elements of the Kalman filter: the state vector and covariance, the system model, the measurement vector and covariance, the measurement model, and the algorithm.

The *state vector* is the set of parameters describing a system, known as *states*, which the Kalman filter estimates. Each state may be constant or time varying. For most navigation applications, the states include the components of position or position error. Velocity, attitude, and navigation sensor error states may also be estimated.

Associated with the state vector is an *error covariance matrix*. This represents the uncertainties in the Kalman filter's state estimates and the degree of correlation between the errors in those estimates. The correlation information is important, as there is not always enough information from the measurements to estimate the Kalman filter states independently, while correlations between errors can also build up over the intervals between measurements. A Kalman filter is an iterative process, so the initial values of the state vector and covariance matrix must be set by the user or determined from another process.

The *system model*, also known as the process model or time-propagation model, describes how the Kalman filter states and error covariance matrix vary



Solid line indicates data flows are always present.

Dotted line indicates data flows are in some applications only.

Figure 3.1 Elements of the Kalman filter. (From: [7]. © 2002 QinetiQ Ltd. Reprinted with permission.)

with time. For example, a position state will vary with time as the integral of a velocity state; the position uncertainty will increase with time as the integral of the velocity uncertainty; and the position and velocity estimation errors will become more correlated. The system model is deterministic for the states as it is based on known properties of the system.³

A state uncertainty should also be increased with time to account for unknown changes in the system that cause the state estimate to go out of date in the absence of new measurement information. These changes may be unmeasured dynamics or random noise on an instrument output. For example, a velocity uncertainty must be increased over time if the acceleration is unknown. This variation in the true values of the states is known as *system noise*, and its statistical properties are usually estimated in advance by the Kalman filter designer.

The *measurement vector* is a set of simultaneous measurements of properties of the system, which are functions of the state vector. Examples include the set of range measurements from a radio navigation system and the difference in navigation solution between an INS under calibration and a reference navigation system. This is the information from which all of the state estimates are derived after initialization. Associated with the measurement vector is a *measurement noise covariance* matrix, which describes the statistics of the noise on the measurements. For many applications, new measurement information is input to the Kalman filter at regular intervals. In other cases, the time interval between measurements can be irregular.

The *measurement model* describes how the measurement vector varies as a function of the true state vector (as opposed to the state vector estimate) in the absence of measurement noise. For example, the velocity measurement difference between an INS under calibration and a reference system is directly proportional to the INS velocity error. Like the system model, the measurement model is deterministic, based on known properties of the system.¹

The *Kalman filter algorithm* uses the measurement vector, measurement model, and system model to maintain optimal estimates of the state vector. It comprises up to 10 steps per iteration. These are shown in Figure 3.2. Steps 1–4 are the system-propagation phase, and steps 5–10 are the measurement-update phase of the Kalman filter.²

The purpose of the system-propagation, or time-propagation, phase is to predict forward the state vector estimate and error covariance matrix from the time of validity of the last measurement set to the time of the current set of measurements using the known properties of the system. So, for example, a position estimate is predicted forward using the corresponding velocity estimate. This provides the Kalman filter's best estimate of the state vector at the current time in the absence of new measurement information. The first two steps calculate the deterministic and noise parts of the system model. The third step, *state propagation*, uses this to bring the state vector estimate up to date. The fourth step, *covariance propagation*, performs the corresponding update to the error covariance matrix, increasing the state uncertainty to account for the system noise.

In the measurement-update phase, the state vector estimate and error covariance are updated to incorporate the new measurement information. Steps 5 and 6 calculate the deterministic and noise parts of the measurement model. The seventh step, *gain computation*, calculates the Kalman gain matrix. This is used to optimally

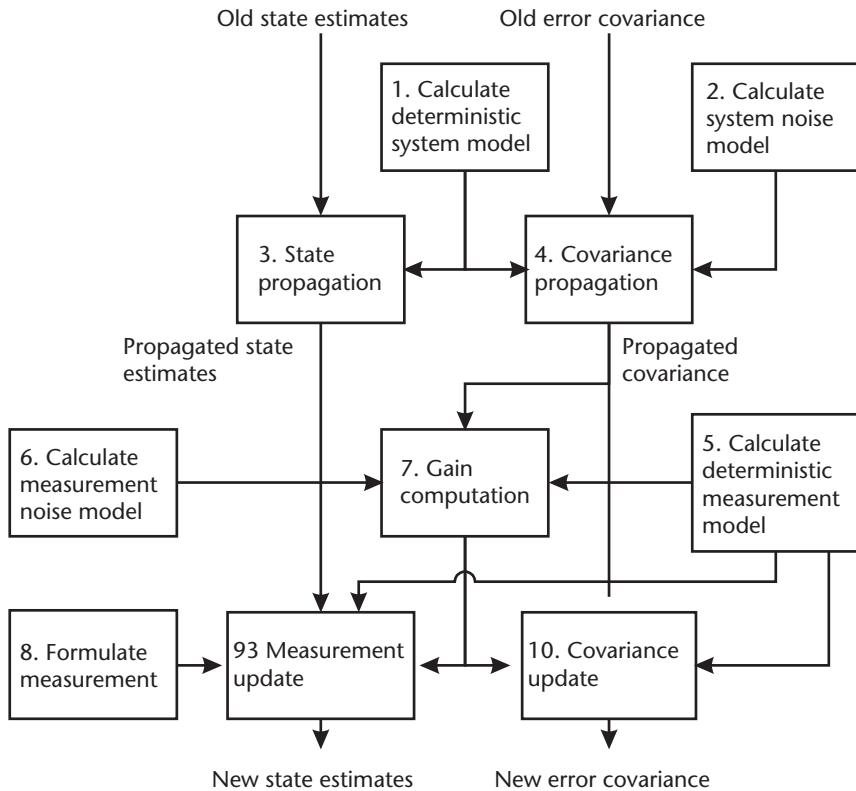


Figure 3.2 Kalman filter algorithm steps.

weight the correction to the state vector according to the uncertainty of the current state estimates and how noisy the measurements are. The eighth step formulates the measurement vector. The ninth step, the *measurement update*, updates the state estimates to incorporate the measurement data weighted with the Kalman gain. Finally, the *covariance update* updates the error covariance matrix to account for the new information that has been incorporated into the state vector estimate from the measurement data.

3.1.2 Kalman Filter Applications

Kalman filter-based estimation techniques have many applications in navigation, including fine alignment and calibration of INS, GNSS navigation, GNSS signal monitoring, INS/GNSS integration, and multisensor integration.¹

For alignment and calibration of an INS, the states estimated are position, velocity, and attitude errors, together with inertial instrument errors, such as accelerometer and gyro biases. The measurements are the position, velocity, and/or attitude differences between the aligning-INS navigation solution and an external reference, such as another INS or GNSS. More details are given in Section 5.5.3 and Chapters 12 and 13.

For GNSS navigation, the states estimated are receiver position, velocity, clock offset, and clock drift, and the measurements are the line-of-sight ranging measure-

ments of each satellite signal made by the receiver. The GNSS navigation filter is described in Section 7.5.2. GNSS signal monitoring uses the same measurements, but because the receiver position and velocity are accurately known and a high-precision receiver clock is used, the time-correlated range errors may be estimated as Kalman filter states. With a network of monitor stations at different locations, the different contributing factors to the range errors may all be estimated as separate states.

For INS/GNSS and multisensor integration, the Kalman filter usually estimates a number of errors of the constituent navigation systems, though in a few architectures, the navigation solution itself is estimated. The measurements processed vary greatly, depending on the type of integration implemented. INS/GNSS integration techniques are described in Chapter 12, with multisensor integration described in Chapter 14.²

3.2 Algorithms and Models

This section presents and derives the Kalman filter algorithm, system model, and measurement model, including open- and closed-loop implementations and a discussion of Kalman filter behavior. Prior to this, error types are discussed and the main Kalman filter parameters defined. Although a Kalman filter may operate continuously, discrete implementations are most common, as these are suited to digital computation. Thus, only the discrete version is presented here.³

3.2.1 Definitions

The time variation of all errors modeled within a Kalman filter is assumed to fall into one of three categories: systematic errors, white noise, and Markov processes. These are shown in Figure 3.3. *Systematic errors* are assumed to be constant—in other words, 100 percent time-correlated, though a Kalman filter’s estimates of these quantities may vary.³

Samples from a *white noise* process taken at different times are uncorrelated. Thus, for a white noise sequence, $w_i(t)$,

$$E(w_i(t_1)w_i(t_2)) = 0 \quad t_1 \neq t_2 \quad (3.1)$$

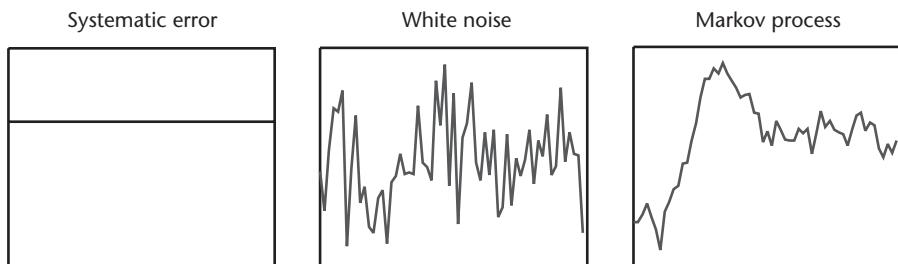


Figure 3.3 Example systematic error, white noise, and Markov process.

where E is the expectation operator. The variance (see Section B.1 in Appendix B) of continuous white noise, integrated over the time interval τ is

$$\begin{aligned}\sigma_{wr}^2 &= E\left(\int_{t-\tau}^t \int_{t-\tau}^t w_i(t') w_i(t'') dt' dt''\right) \\ &= E\left(\int_{t-\tau}^t \int_{t-\tau}^t w_i^2(t') \delta_{t't''} dt' dt''\right) \\ &= \int_{t-\tau}^t n_i^2 dt' \\ &= n_i^2 \tau\end{aligned}\tag{3.2}$$

where δ is the Kronecker delta function and n_i^2 is the power spectral density (PSD) of w_i , which is the variance per unit bandwidth. In general, the PSD is a function of frequency. However, for continuous white noise, it is constant. In a Kalman filter, white noise is assumed to have a zero-mean Gaussian (or normal) distribution (see Section B.3 in Appendix B).

Markov processes are quantities that vary slowly with time. A first-order Markov process may be represented as a function only of its previous value and noise. Where the properties of a Markov process are known, it can be modeled in a Kalman filter. For example, a first-order Markov process with an exponentially decaying auto-correlation function, x_{mi} , is described by

$$\frac{\partial x_{mi}}{\partial t} = -\frac{x_{mi}}{\tau_{mi}} + w_i\tag{3.3}$$

where t is time and τ_{mi} is the correlation time.

A principal assumption of Kalman filters is that the errors of the system being modeled are either systematic, white noise, or Markov processes. They may also be linear combinations or integrals thereof. For example, a random walk process is integrated white noise, while a constant acceleration error leads to a velocity error that grows with time. In a standard Kalman filter, error sources modeled as states are assumed to be systematic errors, Markov processes, or their integrals. All noise sources are assumed to be white, noting that Markov processes have a white noise component. Real navigation system errors do not fall neatly into these categories, but, in many cases, can be approximated to them, provided the modeled errors adequately overbound their real counterparts. A good analogy is that you can fit a square peg into a round hole if you make the hole sufficiently large.¹

The set of parameters estimated by a Kalman filter, known as the *state vector*, is denoted by \mathbf{x} . The Kalman filter estimate of the state vector is denoted $\hat{\mathbf{x}}$, with the caret, $\hat{\cdot}$, also used to indicate other quantities calculated using the state estimates. Estimating absolute properties of the system, such as position, velocity, and attitude

errors, as states is known as a *total-state* implementation. Estimation of the errors in a measurement made by the system, such as INS position, velocity, and attitude errors, as states is known as an *error-state* implementation, also known as a complementary filter. However, a state vector may comprise a mixture of total states and error states.

The *state vector residual*, $\delta\mathbf{x}$, is the difference between the true state vector and the Kalman filter estimates thereof. Thus,

$$\delta\mathbf{x} = \mathbf{x} - \hat{\mathbf{x}} \quad (3.4)$$

In an error-state implementation, the state vector residual represents the errors remaining in the system after the Kalman filter estimates have been used to correct it. The errors in the state estimates are obtained simply by reversing the sign of the state residuals.

The *error covariance matrix*, \mathbf{P} , defines the expectation of the square of the deviation of the state vector estimate from the true value of the state vector. Thus,

$$\mathbf{P} = E((\hat{\mathbf{x}} - \mathbf{x})(\hat{\mathbf{x}} - \mathbf{x})^T) = E(\delta\mathbf{x}\delta\mathbf{x}^T) \quad (3.5)$$

The diagonal elements of \mathbf{P} are the variances of each state estimate, while their square roots are the uncertainties. The off-diagonal elements, the covariances, give the correlations between the errors in the different state estimates. State estimates can become significantly correlated with each other where there is insufficient information from the measurements to estimate those states independently. It is analogous to having a set of simultaneous equations where there are more unknowns than equations. This subject is known as observability and is discussed further in Section 3.2.3.

In an error-state implementation, all state estimates are usually given an initial value of zero. In a total-state implementation, the states may be initialized by the user, by a coarse initialization process, or with the estimates from the previous time the host equipment was used. The initialization values of the covariance matrix are generally determined by the Kalman filter designer. Thus, the state initialization values are a priori estimates, while the initial covariance matrix values indicate the confidence in those estimates.

It is necessary to distinguish between the state vector and error covariance after complete iterations of the Kalman filter and in the intermediate step between propagation and update. Thus, the time-propagated state estimates and covariance are denoted $\hat{\mathbf{x}}_k^-$ and \mathbf{P}_k^- (some authors use $\hat{\mathbf{x}}_k(-)$ and $\mathbf{P}_k(-)$), $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ or $\hat{\mathbf{x}}(k | k - 1)$ and $\mathbf{P}(k | k - 1)$). Their counterparts following the measurement update are denoted $\hat{\mathbf{x}}_k^+$ and \mathbf{P}_k^+ (some authors use $\hat{\mathbf{x}}_k(+)$ and $\mathbf{P}_k(+)$, $\hat{\mathbf{x}}_{k|k}$ and $\mathbf{P}_{k|k}$ or $\hat{\mathbf{x}}(k | k)$ and $\mathbf{P}(k | k)$). The subscript k is commonly used to denote the iteration.

The measurement vector, \mathbf{z} (some authors use \mathbf{y}), is a set of measurements of the properties of the system described by the state vector. This could be a set of range measurements or the difference between two navigation systems' position and velocity solutions. It comprises a deterministic function, $\mathbf{h}(\mathbf{x})$ and noise, \mathbf{w}_m (most authors use \mathbf{v}). Thus,

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{w}_m \quad (3.6)$$

The *measurement innovation*, $\delta\mathbf{z}^-$ (some authors use $\boldsymbol{\nu}$), is the difference between the true measurement vector and that computed from the state vector estimate prior to the measurement update:²

$$\delta\mathbf{z}^- = \mathbf{z} - \mathbf{h}(\hat{\mathbf{x}}^-) \quad (3.7)$$

For example, it could be the difference between an actual set of range measurements and a set predicted using a Kalman filter's position estimate. The *measurement residual*, $\delta\mathbf{z}^+$, is the difference between the true measurement vector and that computed from the updated state vector:

$$\delta\mathbf{z}^+ = \mathbf{z} - \mathbf{h}(\hat{\mathbf{x}}^+) \quad (3.8)$$

Beware that some authors use the term residual to describe the innovation.

The measurement innovations and residuals are a mixture of state estimation errors and measurement errors that are uncorrelated with the state estimates, such as the noise on a set of range measurements. The standard Kalman filter assumes that these measurement errors form a zero-mean Gaussian distribution, uncorrelated in time, and models their standard deviations with the *measurement noise covariance matrix*, \mathbf{R} . This defines the expectation of the square of the measurement noise. Thus,

$$\mathbf{R} = E(\mathbf{w}_m \mathbf{w}_m^T) \quad (3.9)$$

The diagonal terms of \mathbf{R} are the variances of each measurement, and the off-diagonal terms represent the correlation between the different components of the measurement noise. For most navigation applications, the noise on each component of the measurement vector is independent so \mathbf{R} is a diagonal matrix. The rest of the Kalman filter notation is defined as it is used.³

3.2.2 Kalman Filter Algorithm

With reference to Figure 3.2, the Kalman filter algorithm comprises the following steps:¹

1. Calculate the transition matrix, Φ_{k-1} ;
2. Calculate the system noise covariance matrix, \mathbf{Q}_{k-1} ;
3. Propagate the state vector estimate from $\hat{\mathbf{x}}_{k-1}^+$ to $\hat{\mathbf{x}}_k^-$;
4. Propagate the error covariance matrix from \mathbf{P}_{k-1}^+ to \mathbf{P}_k^- ;
5. Calculate the measurement matrix, \mathbf{H}_k ;
6. Calculate the measurement noise covariance matrix, \mathbf{R}_k ;
7. Calculate the Kalman gain matrix, \mathbf{K}_k ;
8. Formulate the measurement, \mathbf{z}_k ;
9. Update the state vector estimate from $\hat{\mathbf{x}}_k^-$ to $\hat{\mathbf{x}}_k^+$;
10. Update the error covariance matrix from \mathbf{P}_k^- to \mathbf{P}_k^+ .

The Kalman filter steps do not have to be implemented strictly in this order, provided the dependencies depicted in Figure 3.2 are respected. Although many

Kalman filters simply alternate the system-propagation and measurement-update phases, other processing cycles are possible, as discussed in Section 3.3.2.

The first four steps comprise the system-propagation phase of the Kalman filter, also known as the system-update, system-extrapolation, prediction, time-update, or time-propagation phase. The system model is derived in Section 3.2.4.

The *transition matrix*, Φ_{k-1} , defines how the state vector changes with time as a function of the dynamics of the system modeled by the Kalman filter. For example, a position error state will vary as the integral of a velocity error state. A different transition matrix is derived for every Kalman filter application as a function of that system. The transition matrix is always a function of the time interval, τ_s , between Kalman filter iterations and is often a function of other parameters. However, the transition matrix is not a function of any of the states in a standard Kalman filter. If the parameters that the transition matrix is a function of vary, then Φ_{k-1} must be calculated on every iteration of the Kalman filter.

The *system noise covariance matrix*, \mathbf{Q}_{k-1} , defines how the uncertainties of the state estimates increase with time due to noise sources in the system modeled by the Kalman filter, such as unmeasured dynamics and instrument noise. It is always a function of the time interval between iterations, τ_s . It is common for the system noise covariance matrix to be diagonal and constant, but this is not always the case.

The state vector estimate is propagated through time using

$$\hat{\mathbf{x}}_k^- = \Phi_{k-1} \hat{\mathbf{x}}_{k-1}^+ \quad (3.10)$$

There are a number of forms of the covariance propagation in use. The simplest is

$$\mathbf{P}_k^- = \Phi_{k-1} \mathbf{P}_{k-1}^+ \Phi_{k-1}^T + \mathbf{Q}_{k-1} \quad (3.11)$$

Note that the first Φ matrix propagates the rows of the error covariance matrix, while the second, Φ^T , propagates the columns. Following this step, each state uncertainty should be either larger or unchanged.

The remaining steps in the Kalman filter algorithm comprise the measurement-update or correction phase. The measurement model is derived in Section 3.2.5.

The *measurement matrix*, \mathbf{H}_k (some authors use \mathbf{M}_k , while \mathbf{G}_k is sometimes used in GNSS navigation filters), defines how the measurement vector varies with the state vector. For example, the range measurements from a radio navigation system vary with the position of the receiver. In a standard Kalman filter, each measurement is assumed to be a linear function of the state vector. Thus,

$$\mathbf{h}(\mathbf{x}_k^-) = \mathbf{H}_k \mathbf{x}_k^- \quad (3.12)$$

In most applications, the measurement matrix varies, so it must be calculated on each iteration of the Kalman filter. In navigation, \mathbf{H}_k is commonly a function of the user kinematics and/or GNSS satellite geometry. The measurement noise

covariance matrix, \mathbf{R}_k , may be assumed constant or modeled as a function of kinematics or signal-to-noise measurements.

The *Kalman gain matrix*, \mathbf{K}_k , is used to determine the weighting of the measurement information in updating the state estimates. It is a function of the ratio of the uncertainty of the true measurement, \mathbf{z}_k , to the uncertainty of the measurement predicted from the state estimates, $\mathbf{H}_k \hat{\mathbf{x}}_k^-$. From (3.6), (3.7), and (3.9), the square of the uncertainty of the true measurement vector is

$$\mathbb{E}((\mathbf{z}_k - \mathbf{H}_k \mathbf{x}_k)(\mathbf{z}_k - \mathbf{H}_k \mathbf{x}_k)^T) = \mathbf{R}_k \quad (3.13)$$

and, from (3.5), the square of the uncertainty of the measurement vector predicted from the state vector is

$$\mathbb{E}((\mathbf{H}_k \hat{\mathbf{x}}_k^- - \mathbf{H}_k \mathbf{x}_k)(\mathbf{H}_k \hat{\mathbf{x}}_k^- - \mathbf{H}_k \mathbf{x}_k)^T) = \mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T \quad (3.14)$$

The Kalman gain matrix is²

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1} \quad (3.15)$$

where $(\)^{-1}$ denotes the inverse of a matrix. Some authors use a fraction notation for matrix inversion; however, this can leave the order of matrix multiplication ambiguous. Note that, as the leading \mathbf{H}_k matrix of (3.14) is omitted in the *numerator* of the variance ratio, the Kalman gain matrix transforms from measurement space to state space, as well as weighting the measurement information. The correlation information in the off-diagonal elements of the \mathbf{P}_k matrix couples the measurement vector to those states that are not directly related via the \mathbf{H}_k matrix. Matrix inversion is discussed in Section A.4 of Appendix A.

Next, the measurement vector, \mathbf{z}_k , must be formulated. In some applications, the measurement vector components are already present in the system modeled by the Kalman filter (e.g., radio navigation range measurements). In other applications, \mathbf{z}_k must be calculated as a function of other system parameters. An example is the navigation solution difference between a system under calibration and a reference system.

For many applications, the measurement innovation, $\delta \mathbf{z}_k^-$, may be calculated directly by applying corrections derived from the state estimates to those parameters of which the measurements are a function. For example, the navigation solution of an INS under calibration may be corrected by the Kalman filter state estimates prior to being differenced with a reference navigation solution.

The state vector is updated with the measurement vector using¹

$$\begin{aligned} \hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \\ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k \delta \mathbf{z}_k^- \end{aligned} \quad (3.16)$$

The measurement innovation, $\delta \mathbf{z}_k^-$, is multiplied by the Kalman gain matrix to obtain a correction to the state vector estimate. Similarly, the error covariance matrix is updated with

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \quad (3.17)$$

As the updated state vector estimate is based on more information, the updated state uncertainties are smaller than before the update.

Figure 3.4 illustrates the Kalman filter data flow.

The algorithm presented here is for an *open-loop implementation* of the Kalman filter, whereby all state estimates are retained in the Kalman filter algorithm. Section 3.2.6 describes the closed-loop implementation, whereby state estimates are fed back to correct the system.²

3.2.3 Kalman Filter Behavior

Figure 3.5 shows how the uncertainty of a well-observed state estimate varies during the initial phase of Kalman filter operation, where the state estimates are converging with their true counterparts. Note that the state uncertainties are the root diagonals of the error covariance matrix, \mathbf{P} . Initially, when the state uncertainties are large, the Kalman gain will be large, weighting the state estimates toward the new measurement data. The Kalman filter estimates will change quickly as they converge with the true values of the states, so the state uncertainty will drop rapidly. However, assuming a constant measurement noise covariance, \mathbf{R} , this causes the Kalman gain to drop, weighting the state estimates more toward their previous values. This

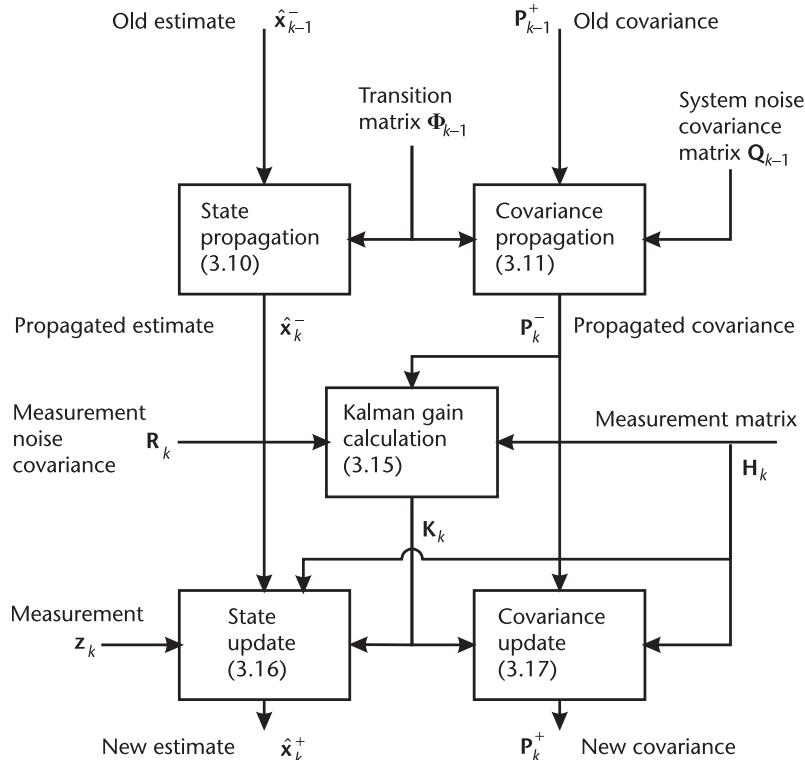


Figure 3.4 Kalman filter data flow.

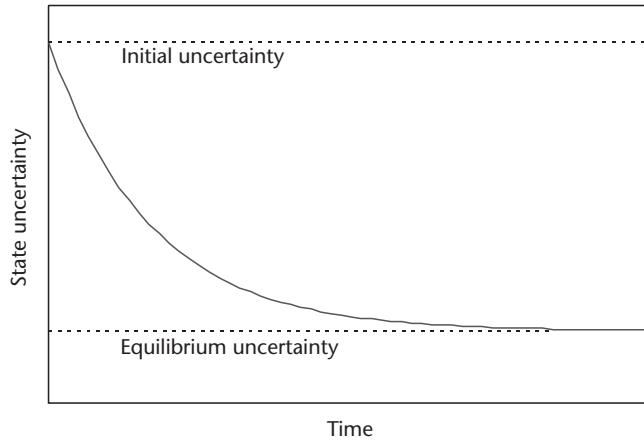


Figure 3.5 Kalman filter state uncertainty during convergence.

reduces the rate at which the states change, so the reduction in the state uncertainty slows. Eventually, the Kalman filter will approach equilibrium, whereby the decrease in state uncertainty with each measurement update is matched by the increase in uncertainty due to system noise. At equilibrium, the state estimates may still vary, but the level of confidence in those estimates, reflected by the state uncertainty, will be more or less fixed.

The rate at which a state estimate converges, if at all, depends on the *observability* of that state. The Kalman filter's measurement model is analogous to a set of simultaneous equations where the states are the unknowns to be found, the measurements are the known quantities, and the measurement matrix, \mathbf{H} , provides the coefficients of the states. Therefore, on a single iteration, the Kalman filter cannot completely observe more states than there are components of the measurement vector, merely linear combinations of those states. However, if the measurement matrix changes over time or there is a time-dependent relationship between states through the transition matrix, Φ , then it is possible, over time, to observe more states than there are measurement components. The error covariance matrix, \mathbf{P} , records the correlations between the state estimates as well as their uncertainties. A good example in navigation is determination of velocity from the rate of change of position.

The observability of many parameters is dynamics dependent. For example, the attitude errors and accelerometer biases of an INS are not separately observable at constant attitude, but they are after a change in attitude as this changes the relationship between the states in the system model. Observation of gyro errors and higher order accelerometer errors requires much higher dynamics. However, if two states have the same effect on the measurements and vary with time and dynamics in the same way, they will never be separately observable, so they should be combined to avoid wasting processing resources.

As well as the relationships with the measurements and the other states, the rate of convergence of a Kalman filter state depends on the measurement sampling rate, the magnitude and correlation properties of the measurement noise, and the level of system noise. This is known as *stochastic observability*. System and

measurement noise can mask the effects of those states that only have a small impact on the measurements, making those states effectively unobservable.

3.2.4 System Model

To propagate the state vector estimate, $\hat{\mathbf{x}}$, and error covariance, \mathbf{P} , forward in time, it is necessary to know how those states vary with time. This is the function of the system model. This section shows how the Kalman filter system propagation equations, (3.10) and (3.11), may be obtained from a model of the state dynamics.

An assumption of the Kalman filter is that the time derivative of each state is a linear function of the other states and of white noise sources. Thus, the true state vector, $\mathbf{x}(t)$, at time, t , of any Kalman filter is described by the following dynamic model:¹

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}_s(t) \quad (3.18)$$

where $\mathbf{w}_s(t)$ is the system noise vector, $\mathbf{F}(t)$ is the system matrix (some authors use \mathbf{A}), and $\mathbf{G}(t)$ is the system noise distribution matrix (some authors use $\mathbf{\Gamma}$). The system noise vector comprises a number of independent random noise sources, each assumed to have a zero-mean Gaussian distribution. $\mathbf{F}(t)$ and $\mathbf{G}(t)$ are always known functions. To determine the system model, these functions must be derived from the known properties of the system.²

A simple example is a two-state Kalman filter, estimating the position and velocity along a single axis in a nonrotating frame. As the acceleration is not estimated, this becomes the system noise. Thus, the state vector and system noise are

$$\mathbf{x} = \begin{pmatrix} x_{ib,x}^i \\ v_{ib,x}^i \end{pmatrix}, \quad w_s = a_{ib,x}^i \quad (3.19)$$

The state dynamics are simply

$$\dot{x}_{ib,x}^i = v_{ib,x}^i, \quad \dot{v}_{ib,x}^i = a_{ib,x}^i \quad (3.20)$$

Substituting (3.19) and (3.20) into (3.18) gives the system matrix and system noise distribution matrix:

$$\mathbf{F} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (3.21)$$

noting that, in this case, neither matrix is a function of time.

To obtain an estimate, the expectation operator, E , is applied. The expectation value of the true state vector, $\mathbf{x}(t)$, is the estimated state vector, $\hat{\mathbf{x}}(t)$. The expectation value of the system noise vector, $\mathbf{w}_s(t)$, is zero, as the noise is assumed to be of zero mean. $\mathbf{F}(t)$ and $\mathbf{G}(t)$ are assumed to be known functions and thus commute with the expectation operator. Hence, taking the expectation of (3.18) gives¹

$$E(\dot{\mathbf{x}}(t)) = \frac{\partial}{\partial t} \hat{\mathbf{x}}(t) = \mathbf{F}(t) \hat{\mathbf{x}}(t) \quad (3.22)$$

Solving (3.22) gives the state vector estimate at time t as a function of the state vector estimate at time $t - \tau_s$:

$$\hat{\mathbf{x}}(t) = \exp \left(\int_{t-\tau_s}^t \mathbf{F}(t') dt' \right) \hat{\mathbf{x}}(t - \tau_s) \quad (3.23)$$

In the discrete Kalman filter, the state vector estimate is modeled as a linear function of its previous value, coupled by the transition matrix, Φ_{k-1} , with (3.10) repeated here:

$$\hat{\mathbf{x}}_k^- = \Phi_{k-1} \hat{\mathbf{x}}_{k-1}^+$$

The discrete and continuous forms of the Kalman filter are equivalent, with $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}(t)$ and $\hat{\mathbf{x}}_{k-1} = \hat{\mathbf{x}}(t - \tau_s)$. So, substituting (3.23) into (3.10),

$$\Phi_{k-1} \approx \exp(\mathbf{F}_{k-1} \tau_s) \quad (3.24)$$

where, assuming data is available at times $t - \tau_s$ and t , but not at intervening intervals, the system matrix, \mathbf{F}_{k-1} , can be calculated either as $\frac{1}{2}(\mathbf{F}(t - \tau_s) + \mathbf{F}(t))$ or by taking the mean of the parameters of \mathbf{F} at times $t - \tau_s$ and t and making a single calculation of \mathbf{F} . In general, (3.24) cannot be computed directly; the exponent of the matrix is not the matrix of the exponents of its components.² Numerical methods are available [8], but these are computationally intensive where the matrices are large. Therefore, the transition matrix is usually computed as a power-series expansion of the system matrix, \mathbf{F} , and propagation interval, τ_s :

$$\Phi_{k-1} = \sum_{r=0}^{\infty} \frac{\mathbf{F}_{k-1}^r \tau_s^r}{r!} = \mathbf{I} + \mathbf{F}_{k-1} \tau_s + \frac{1}{2} \mathbf{F}_{k-1}^2 \tau_s^2 + \frac{1}{6} \mathbf{F}_{k-1}^3 \tau_s^3 + \dots \quad (3.25)$$

The Kalman filter designer must decide where to truncate the power-series expansion, depending on the likely magnitude of the states, the length of the propagation interval, and the available error margins. With a shorter propagation interval, a given accuracy may be attained with a shorter truncation. Different truncations may be applied to different terms, and exact solutions may be available for some elements of the transition matrix. In some cases, such as the example in (3.21), \mathbf{F}^2 is zero, so the first-order solution, $\mathbf{I} + \mathbf{F}_{k+1} \tau_s$, is exact.

The true state vector can be obtained as a function of its previous value, \mathbf{x}_{k-1} , by integrating (3.18) between times $t - \tau_s$ and t under the approximation that $\mathbf{F}(t)$ and $\mathbf{G}(t)$ are constant over the integration interval and substituting (3.24):

$$\mathbf{x}_k = \Phi_{k-1} \mathbf{x}_{k-1} + \int_{t-\tau_s}^t \exp(\mathbf{F}_{k-1}(t-t')) \mathbf{G}_{k-1} \mathbf{w}_s(t') dt' \quad (3.26)$$

Note that, as system noise is introduced throughout the propagation interval, it is subject to state propagation via \mathbf{F} for the remainder of that propagation interval. The system noise distribution matrix, \mathbf{G}_{k-1} , is calculated in a similar manner to \mathbf{F}_{k-1} , either as $\frac{1}{2}(\mathbf{G}(t - \tau_s) + \mathbf{G}(t))$ or by taking the mean of the parameters of \mathbf{G} at times $t - \tau_s$ and t and making a single calculation of \mathbf{G} .

From (3.5), the error covariance matrix before and after the time propagation is

$$\begin{aligned} \mathbf{P}_k^- &= E[(\hat{\mathbf{x}}_k^- - \mathbf{x}_k)(\hat{\mathbf{x}}_k^- - \mathbf{x}_k)^T] \\ \mathbf{P}_k^+ &= E[(\hat{\mathbf{x}}_k^+ - \mathbf{x}_k)(\hat{\mathbf{x}}_k^+ - \mathbf{x}_k)^T] \end{aligned} \quad (3.27)$$

Subtracting (3.26) from (3.10),

$$\hat{\mathbf{x}}_k^- - \mathbf{x}_k = \Phi_{k-1}(\hat{\mathbf{x}}_k^+ - \mathbf{x}_k) - \int_{t-\tau_s}^t \exp(\mathbf{F}_{k-1}(t-t')) \mathbf{G}_{k-1} \mathbf{w}_s(t') dt' \quad (3.28)$$

The errors in the state estimates are uncorrelated with the system noise, so

$$E[(\hat{\mathbf{x}}_k^\pm - \mathbf{x}_k)\mathbf{w}_s^T(t)] = 0, \quad E[\mathbf{w}_s(t)(\hat{\mathbf{x}}_k^\pm - \mathbf{x}_k)^T] = 0 \quad (3.29)$$

Therefore, substituting (3.28) and (3.29) into (3.27) gives the exact form of the covariance propagation equation:

$$\begin{aligned} \mathbf{P}_k^- &= \Phi_{k-1} \mathbf{P}_{k-1}^+ \Phi_{k-1}^T \\ &+ E \left[\int_{t-\tau_s}^t \int_{t-\tau_s}^t \exp(\mathbf{F}_{k-1}(t-t')) \mathbf{G}_{k-1} \mathbf{w}_s(t') \mathbf{w}_s^T(t'') \mathbf{G}_{k-1}^T \exp(\mathbf{F}_{k-1}^T(t-t'')) dt' dt'' \right] \end{aligned} \quad (3.30)$$

The integral term in (3.30) is usually approximated. The simplest version is obtained by neglecting the time propagation of the system noise over an iteration of the discrete filter, with (3.11) repeated here:¹

$$\mathbf{P}_k^- = \Phi_{k-1} \mathbf{P}_{k-1}^+ \Phi_{k-1}^T + \mathbf{Q}_{k-1}$$

where, the system noise covariance matrix, \mathbf{Q}_{k-1} , is defined as

$$\mathbf{Q}_{k-1} = \mathbf{G}_{k-1} \mathbf{E} \left(\int_{t-\tau_s}^t \int_{t-\tau_s}^t \mathbf{w}_s(t') \mathbf{w}_s^T(t'') dt' dt'' \right) \mathbf{G}_{k-1}^T \quad (3.31)$$

Alternatively, (3.30) may be approximated to first order in $\Phi_{k-1} \mathbf{Q}_{k-1} \Phi_{k-1}^T$, giving

$$\mathbf{P}_k^- = \Phi_{k-1} \left(\mathbf{P}_{k-1}^+ + \frac{1}{2} \mathbf{Q}_{k-1} \right) \Phi_{k-1}^T + \frac{1}{2} \mathbf{Q}_{k-1} \quad (3.32)$$

Returning to the example in (3.19)–(3.21) of the two-state Kalman filter estimating position and velocity along a single axis, if the acceleration is approximated as white noise, (3.2) can be applied, giving a system noise covariance matrix of

$$\mathbf{Q} = \begin{pmatrix} 0 & 0 \\ 0 & n_a^2 \tau_s \end{pmatrix} \quad (3.33)$$

where n_a^2 is the PSD of the acceleration.

Time-correlated system noise is discussed in Section 3.4.2.

3.2.5 Measurement Model

In order to update the state vector estimate with a set of measurements, it is necessary to know how the measurements vary with the states. This is the function of the measurement model. This section presents the derivation of the Kalman filter measurement-update equations, (3.15), (3.16), and (3.17) from the measurement model.

In a standard Kalman filter, the measurement vector, $\mathbf{z}(t)$, is modeled as a linear function of the true state vector, $\mathbf{x}(t)$, and the white noise sources, $\mathbf{w}_m(t)$. Thus,

$$\mathbf{z}(t) = \mathbf{H}(t) \mathbf{x}(t) + \mathbf{w}_m(t) \quad (3.34)$$

where $\mathbf{H}(t)$ is the measurement matrix and is determined from the known properties of the system. For example, if the state vector comprises the position error of a dead reckoning system, such as an INS, and the measurement vector comprises the difference between the dead reckoning system's position solution and that of a positioning system, such as GNSS, then the measurement matrix is simply the identity matrix. If the measurements are taken at discrete intervals, (3.34) becomes

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{w}_{mk} \quad (3.35)$$

Following the measurement, the new optimal estimate of the state vector is a linear combination of the measurement and the previous estimation, so¹

$$\hat{\mathbf{x}}_k^+ = \mathbf{K}_k \mathbf{z}_k + \mathbf{K}'_k \hat{\mathbf{x}}_k^- \quad (3.36)$$

where \mathbf{K}_k and \mathbf{K}'_k are weighting functions to be determined.² Substituting in (3.35),

$$\hat{\mathbf{x}}_k^+ = \mathbf{K}_k \mathbf{H}_k \mathbf{x}_k + \mathbf{K}_k \mathbf{w}_{mk} + \mathbf{K}'_k \hat{\mathbf{x}}_k^- \quad (3.37)$$

A Kalman filter is an unbiased estimation algorithm, so the expectation of the state vector residual, $\delta\mathbf{x}$, is zero. So, from (3.4), the expectation value of both old and new state vector estimates is the true state vector, \mathbf{x}_k . The expectation of the measurement noise, \mathbf{w}_{mk} , is also zero. Thus, taking the expectation of (3.37) gives

$$\mathbf{K}'_k = \mathbf{I} - \mathbf{K}_k \mathbf{H}_k \quad (3.38)$$

Substituting this into (3.36) gives the state vector update equation, with (3.16) repeated here:

$$\begin{aligned}\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \\ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k \delta\mathbf{z}_k^-\end{aligned}$$

Substituting (3.35) into (3.16) and subtracting the true state vector,

$$\hat{\mathbf{x}}_k^+ - \mathbf{x}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)(\hat{\mathbf{x}}_k^- - \mathbf{x}_k) + \mathbf{K}_k \mathbf{w}_{mk} \quad (3.39)$$

The error covariance matrix after the measurement update, \mathbf{P}_k^+ , is then obtained by substituting this into (3.27), giving

$$\begin{aligned}\mathbf{P}_k^+ &= \mathbb{E} \left[(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{w}_{mk} (\hat{\mathbf{x}}_k^- - \mathbf{x}_k)^T (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T \right. \\ &\quad \left. + (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) (\hat{\mathbf{x}}_k^- - \mathbf{x}_k) \mathbf{w}_{mk}^T \mathbf{K}_k^T + \mathbf{K}_k \mathbf{w}_{mk} \mathbf{w}_{mk}^T \mathbf{K}_k^T \right] \quad (3.40)\end{aligned}$$

The error in the state vector estimates is uncorrelated with the measurement noise so,¹

$$\mathbb{E}[(\hat{\mathbf{x}}_k^\pm - \mathbf{x}_k) \mathbf{w}_{mk}^T] = 0 \quad \mathbb{E}[\mathbf{w}_{mk} (\hat{\mathbf{x}}_k^\pm - \mathbf{x}_k)^T] = 0 \quad (3.41)$$

\mathbf{K}_k and \mathbf{H}_k commute with the expectation operator, so substituting (3.41) and (3.9) into (3.40) gives

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T \quad (3.42)$$

noting that the measurement noise covariance matrix, \mathbf{R}_k , is defined by (3.9). This equation is known as the Joseph form of the covariance update.

Now, the criterion for optimally selecting the weighting function, \mathbf{K}_k , is the minimization of the error in the estimate $\hat{\mathbf{x}}_k^+$. The variances of the state estimates are given by the diagonal elements of the error covariance matrix. It is therefore necessary to minimize the trace of \mathbf{P}_k^+ (see Section A.2 in Appendix A) with respect to \mathbf{K}_k :

$$\frac{\partial}{\partial \mathbf{K}_k} [\text{tr}(\mathbf{P}_k^+)] = 0 \quad (3.43)$$

Substituting in (3.42) and applying the matrix relation

$$\frac{\partial}{\partial \mathbf{A}} [\text{trace}(\mathbf{ABA}^T)] = 2\mathbf{AB}$$

gives

$$-2(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- \mathbf{H}_k^T + 2\mathbf{K}_k \mathbf{R}_k = 0 \quad (3.44)$$

Rearranging (3.15),²

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

As explained in [2], this result is independent of the units and/or scaling of the states.

By substituting (3.15) into (3.42), the error covariance update equation may be simplified to (3.17):¹

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-$$

Where the measurement noise covariance, \mathbf{R}_k , is a diagonal matrix, the components of the measurement vector may be processed as a series of scalars rather than a vector, avoiding the matrix inversion in the calculation of the Kalman gain, \mathbf{K}_k , which can be processor intensive. This is sometimes known as sequential processing. However, the covariance update, (3.17), imposes a lower processor load if all measurements are handled together. Which approach is more computationally efficient depends on the relative size of the state and measurement vectors. Processing the measurement vector as a series of subvectors is also an option.²

Returning to the simple example at the beginning of the subsection, a Kalman filter estimates INS position error using the INS-GNSS position solution difference as the measurement, so the measurement matrix, \mathbf{H} , is the identity matrix. The problem may be simplified further if all components of the measurement have independent noise of standard deviation, σ_z , and the state estimates are uncorrelated and each have an uncertainty of σ_x . This gives

$$\mathbf{H}_k = \mathbf{I}_3, \quad \mathbf{R}_k = \sigma_z^2 \mathbf{I}_3, \quad \mathbf{P}_k^- = \sigma_x^2 \mathbf{I}_3 \quad (3.45)$$

Substituting this into (3.15), the Kalman gain matrix for this example is

$$\mathbf{K}_k = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_z^2} \mathbf{I}_3 \quad (3.46)$$

From (3.16) and (3.17), the state estimates and error covariance are then updated using

$$\hat{\mathbf{x}}_k^+ = \frac{\sigma_z^2 \hat{\mathbf{x}}_k^- + \sigma_x^2 \mathbf{z}_k}{\sigma_x^2 + \sigma_z^2} \quad (3.47)$$

$$\mathbf{P}_k^+ = \frac{\sigma_z^2}{\sigma_x^2 + \sigma_z^2} \mathbf{P}_k^- = \frac{\sigma_x^2 \sigma_z^2}{\sigma_x^2 + \sigma_z^2} \mathbf{I}_3$$

3.2.6 Closed-Loop Kalman Filter

In many navigation applications, such as **INS alignment and calibration**, the true system model is not linear (i.e., the time differential of the state vector varies with terms to second order and higher in the state vector elements). A linear system model is a constraint of the standard Kalman filter design. However, it is often possible to neglect these higher-order terms and still obtain a practically useful Kalman filter. The larger the values of the states estimated, the poorer a given linearity approximation will be.¹

A common technique for getting the best performance out of an error-state Kalman filter with a linearity approximation applied to the system model is the *closed-loop* implementation. Here, the errors estimated by the Kalman filter are fed back every iteration, or at regular intervals, to correct the system itself, zeroing the Kalman filter states in the process. This feedback process keeps the Kalman filter states small, minimizing the effect of neglecting higher order products of states in the system model. Conversely, in the *open-loop* implementation, where there is no feedback, the states will generally get larger as time progresses.

The best stage in the Kalman filter algorithm to feedback the state estimates is immediately after the measurement update. This produces zero state estimates at the start of the state propagation, (3.10), enabling this stage to be omitted completely. The error covariance matrix, \mathbf{P} , is unaffected by the feedback process as the same amount is added to or subtracted from both the true and estimated states, so error covariance propagation, (3.11), is still required.

The closed-loop and open-loop implementations of the Kalman filter may be mixed such that some state estimates are fed back as corrections, whereas others are not. This configuration is useful for applications where feeding back states is desirable, but some states cannot be fed back as there is no way of applying them as corrections to the system. In designing such a Kalman filter, care must be taken in implementing the state propagation as for some of the fed-back states, \mathbf{x}_k^- may be nonzero due to coupling with non-fed-back states through the system model.

Where a full closed-loop Kalman filter is implemented (i.e., with feedback of every state estimate at every iteration), $\mathbf{H}_k \hat{\mathbf{x}}_k^-$ is zero, so the measurement, \mathbf{z}_k , and measurement innovation, $\delta \mathbf{z}_k^-$, are the same.

In navigation, closed-loop Kalman filters are common for the integration, alignment, and calibration of low-grade INS and may also be used for correcting GNSS receiver clocks.²

3.3 Implementation Issues

This section discusses the implementation issues that must be considered in designing a practical Kalman filter. These include tuning and stability, efficient algorithm design, numerical issues, and synchronization. An overall design process is also recommended. Detection of erroneous measurements and biased state estimates is discussed in Chapter 15.

3.3.1 Tuning and Stability

The tuning of the Kalman filter is the selection by the designer or user of values for three matrices. These are the system noise covariance matrix, \mathbf{Q}_k , the measurement noise covariance matrix, \mathbf{R}_k , and the initial values of the error covariance matrix, \mathbf{P}_0^+ . It is important to select these parameters correctly. If the values selected are too small, the actual errors in the Kalman filter estimates will be much larger than the state uncertainties obtained from \mathbf{P} . Conversely, if the values selected are too large, the reported uncertainties will be too large.³ These can cause an external system that uses the Kalman filter estimates to apply the wrong weighting to them.

However, the critical parameter in Kalman filtering is the ratio of the error and measurement noise covariance matrices, \mathbf{P}_k^- and \mathbf{R}_k , as they determine the Kalman gain, \mathbf{K}_k . Figure 3.6 illustrates this. If \mathbf{P}/\mathbf{R} is underestimated, the Kalman gain will be too small and state estimates will converge with their true counterparts more slowly than necessary. The state estimates will also be slow to respond to changes in the system.¹

Conversely, if \mathbf{P}/\mathbf{R} is overestimated, the Kalman gain will be too large. This will bias the filter in favor of more recent measurements, which may result in unstable or biased state estimates due to the measurement noise having too great an influence on them. Sometimes, the state estimates can experience positive feedback of the measurement noise through the system model, causing them to rapidly diverge from their truth counterparts.²

In an ideal Kalman filter application, tuning the noise models to give consistent estimation errors and uncertainties will also produce stable state estimates that track their true counterparts. However, in practice, it is often necessary to tune the filter to give 1σ state uncertainties substantially larger (two or three times is typical) than the corresponding error standard deviations in order to maintain

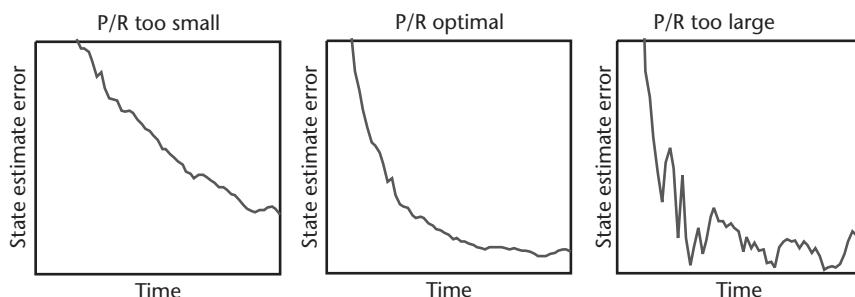


Figure 3.6 Kalman filter error propagation for varying \mathbf{P}/\mathbf{R} ratio.

stability. This is because the Kalman filter's model of the system is only an approximation of the real system.

There are a number of sources of approximation in a Kalman filter. Smaller error states are often neglected due to observability problems or processing-capacity limitations. The system and/or measurement models may have to be approximated in order to meet the linearity requirements of the Kalman filter equations. The stochastic properties of slowly time-varying states are often oversimplified. Nominally constant states may also vary slowly with time (e.g., due to temperature or pressure changes). Allowing state uncertainties to become very small can also precipitate numerical problems (see Section 3.3.3). Therefore, it is advisable to model system noise on all states. Alternatively, lower limits to the state uncertainties may be maintained. Finally, the Kalman filter assumes that all noise sources are white, whereas, in practice, they will exhibit some time correlation due to band-limiting effects. Therefore, to overcome the limitations of the Kalman filter model, sufficient noise must be modeled to overbound the real system's behavior. By analogy, to fit the “square peg” of the real world problem into the “round hole” of the Kalman filter model, the hole must be widened to accommodate the edges of the peg.

For most applications, manufacturers' specifications and laboratory test data may be used to determine suitable values for the initial error covariance and system noise covariance matrices. However, the measurement noise covariance tends to be more nebulous. It can be difficult to separate out measurement noise from system effects in an analysis of the measurement stream. It may also be necessary to exaggerate R in order to account for time correlation in the measurement noise due to band-limiting or synchronization errors. Therefore, a good tuning philosophy is to fix P_0^+ and Q_k and then vary R_k by trial and error to find the smallest value that gives stable state estimates. If this doesn't give satisfactory performance, P_0^+ and Q_k can also be varied. Automatic real-time tuning techniques are discussed in Section 3.4.3.

Tuning a Kalman filter is essentially a tradeoff between convergence rate and stability. However, it is important to note that the convergence rate can also affect the long-term accuracy, as this is reached once the convergence rate matches the rate at which the true states change due to noise effects. For some Kalman filtering applications, integrity monitoring techniques (Chapter 15) can be used to detect and remedy state instability, in which case the tuning may be selected to optimize convergence.³

3.3.2 Algorithm Design

The processing load for implementation of a Kalman filter depends on the number of components of the state vector, n , measurement vector, m , and system noise vector, l , as shown in Table 3.1. Where the number of states is large, the covariance propagation and update require the largest processing capacity. However, where the measurement vector is larger than the state vector, the Kalman gain calculation has the largest impact on processor load.

In moving from a theoretical to a practical Kalman filter, a number of modifications can be made to improve the processing efficiency without significantly

Table 3.1 Multiplications Required by Kalman Filter Processes

Kalman Filter Process	Equation	Multiplications Required
System-propagation phase		
State propagation	(3.10)	n^2
Covariance propagation	(3.11)	$2n^3$
System noise distribution matrix computation	(3.31)	$2nl$
Measurement-update phase		
Kalman gain calculation	(3.15)	$2mn^2 + m^2n$
Matrix inversion		$\sim m^3$
State vector update	(3.16)	$2mn$
Covariance update	(3.17)	$mn^2 + n^3$

impacting on performance. For example, many elements of the transition, Φ_k , and measurement, \mathbf{H}_k , matrices are zero, so it is more efficient to use sparse matrix multiplication routines that only multiply the nonzero elements. However, there is a tradeoff between processing efficiency and algorithm complexity, with more complex algorithms taking longer to develop, code, and debug.³

Another option takes advantage of the error covariance matrix, P_k , being symmetric about the diagonal. By computing only the diagonal elements and either the upper or lower triangle, the computational effort required to propagate and update the covariance matrix may be almost halved.

Sparse matrix multiplication cannot be used for the matrix inversion within the Kalman gain calculation or in updating the covariance, (3.17), other than for computing $\mathbf{K}_k \mathbf{H}_k$. Consequently, the measurement-update phase of the Kalman filter will always require more computational capacity than the system-propagation phase. The interval between measurement updates may be limited by processing power. It may also be limited by the rate at which measurements are available or by the correlation time of the measurement noise. In any case, the measurement-update interval can sometimes be too large to calculate the transition matrix, Φ_k , over, due to the need for the power-series expansion of $F\tau_s$ in (3.25) to converge. However, the different phases of the Kalman filter do not have to be iterated at the same rate. The system propagation may be iterated at a faster rate than the measurement update, reducing the propagation interval, τ_s . Similarly, if a measurement update cannot be performed due to lack of valid data, the system propagation can still go ahead. The update rate for a given measurement stream should not be faster than the system-propagation rate.

The Kalman filter equations involving the covariance matrix, P , impose a much higher computational load than those involving the state vector, x . However, the accuracy requirement for the state vector is higher, particularly for the open-loop Kalman filter, requiring a shorter propagation interval to maximize the transition matrix accuracy. Therefore, it is sometimes more efficient to iterate the state vector propagation, (3.10), at a higher rate than the error covariance propagation, (3.11).

Where the measurement update interval that processing capacity allows is much greater than the noise correlation time of the measurement stream, the noise on the measurements can be reduced by time averaging. In this case, the measurement innovation, δz^- , is calculated at a faster rate and averaged measurement innovations are used to update the state estimates, \hat{x} , and covariance, P , at the rate allowed by

the processing capacity. Where the measurements, \mathbf{z} , rather than the measurement innovations, are averaged, the measurement matrix, \mathbf{H} , must be modified to account for the state propagation over the averaging interval [9]. Measurement averaging is also known as prefiltering.

Altogether, a Kalman filter algorithm may have four different iteration rates for the state propagation, (3.10), error covariance propagation, (3.11), measurement accumulation, and measurement update, (3.15)–(3.17). Figure 3.7 presents an example illustration. Furthermore, different types of measurement input to the same Kalman filter, such as position and velocity or velocity and attitude, may be accumulated and updated at different rates.³

3.3.3 Numerical Issues

When a Kalman filter is implemented on a computer, the precision is limited by the number of bits used to store and process each parameter. The fewer bits used, the larger the rounding errors on each computation will be. The effect of rounding errors on many of the state estimates is corrected by the Kalman filter's measurement update process. However, there are no corresponding corrections to the error covariance matrix, \mathbf{P} . The longer the Kalman filter has been running and the higher the iteration rate, the greater the distortion of the matrix. This distortion manifests as breakage of the symmetry about the diagonal and can even produce negative diagonal elements, which represent imaginary uncertainty. Small errors in the \mathbf{P} matrix are relatively harmless. However, large \mathbf{P} -matrix errors distort the Kalman gain matrix, \mathbf{K} . Gains that are too small produce unresponsive state estimates while gains that are too large can produce unstable, oscillatory state estimates. If an element of the Kalman gain matrix is the wrong sign, a state estimate is liable to diverge away from truth. Extreme covariance matrix distortion can also cause software crashes. Thus, the Kalman filter implementation must be designed to minimize computational errors in the error covariance matrix. In particular, \mathbf{P} must remain positive definite (i.e., retain a positive determinant).

The simplest and most reliable way of minimizing rounding errors is to use high-precision arithmetic—for example, implementing the Kalman filter with double-precision (64-bit) arithmetic as opposed to single precision (32 bit). However, this increases the processing load unless a 64-bit processor is used. Double-precision arithmetic may be used for part of the Kalman filter only. However, the most sensitive step, the covariance measurement update, (3.17), is also generally the most computationally intensive.

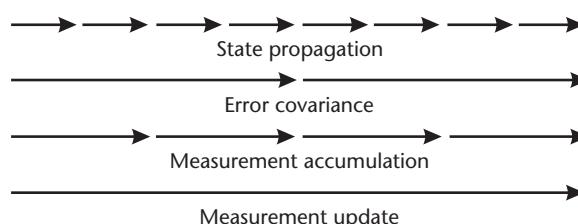


Figure 3.7 Example Kalman filter iteration rates.

There are several other methods for reducing the effect of rounding errors. The Kalman filter states may be scaled so that all state uncertainties are of a similar order of magnitude in numerical terms, effectively reducing the dynamic range of the error covariance matrix. This is particularly important where fixed-point, as opposed to floating-point, arithmetic is used. P-matrix symmetry can be forcibly maintained by averaging the matrix with its transpose after each system propagation and measurement update. Lower limits can also be applied to the state uncertainties.

Another way of minimizing the effects of rounding errors is to modify the Kalman filter algorithm. The Joseph form of the covariance update replaces (3.17) with (3.42). It has greater symmetry than the standard form, but requires more than twice the processing capacity. A common approach is covariance factorization. These techniques effectively propagate \sqrt{P} rather than P , reducing the dynamic range by a factor of 2 so that rounding errors have less impact. A number of factorization techniques are reviewed in [10], but the most commonly used is the Bierman-Thornton or UDU method [3, 11].

There is a particular risk of numerical problems at the first measurement update following initialization when the initial uncertainties are large and the measurement noise covariance is small. This is because there can be a very large change in the error covariance matrix, with the covariance update comprising the multiplication of very large numbers with very small numbers. If problems occur, the initial state uncertainties should be set artificially small. As long as the values used are still larger than those expected after convergence, the state uncertainties will be corrected as the Kalman filter converges [4].

3.3.4 Handling Data Lags

Different types of navigation system exhibit different data lags between the time at which sensor measurements are taken, known as the *time of validity*, and the time when a navigation solution based on those measurements is output. There may also be a communication delay between the navigation system and the Kalman filter processor. Where Kalman filter measurements compare the outputs of two different navigation systems, it is important to ensure that those outputs correspond to the same time of validity. Otherwise, differences in the navigation system outputs due to the time lag between them will be falsely attributed by the Kalman filter to the states, corrupting the estimates of those states. The greater the level of dynamics encountered, the larger the impact of a given time-synchronization error will be. Poor time synchronization can be mitigated by using very low gains in the Kalman filter; however, it is better to synchronize the measurement data.

Data synchronization requires the outputs from the faster responding system, such as an INS, to be stored. Once an output is received from the slower system, such as a GNSS receiver, an output from the faster system with the same time of validity is retrieved from the store and used to form a synchronized measurement input to the Kalman filter. Figure 3.8 illustrates the architecture. It is usually better to interpolate the data in the store rather than use the nearest point in time. Data-lag compensation is more effective where all data is time-tagged, enabling precise synchronization. Where time tags are unavailable, data lag compensation may

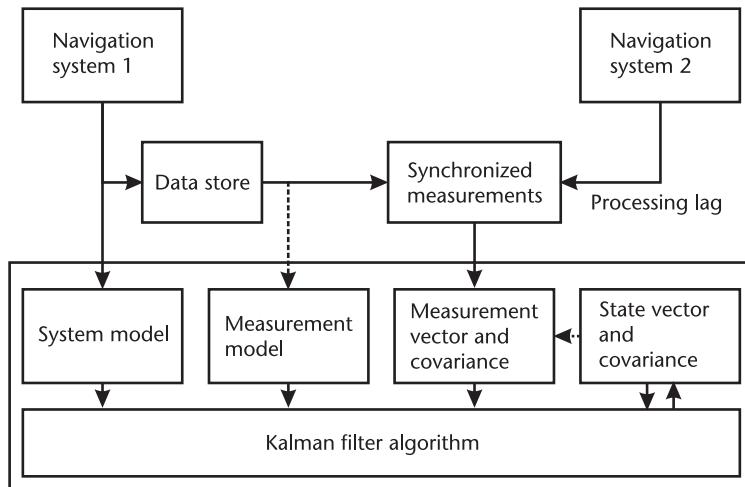


Figure 3.8 Data synchronization (open-loop Kalman filter).

operate using an assumed average time delay, provided this is known to within about 10 ms and the actual lag does not vary by more than about ± 100 ms.³

The system-propagation phase of the Kalman filter usually uses data from the faster responding navigation system. Consequently, the state estimates may be propagated to a time ahead of the measurement time of validity. The optimal solution is to post-multiply the measurement matrix, H , by a transition matrix that propagates between the state and measurement times of validity. However, this is usually unnecessary in practice. Another option is simply to limit the extent of the system propagation to the measurement time of validity.

Where closed-loop correction of the navigation system(s) under calibration by the Kalman filter is used, data-delay compensation introduces a delay in applying the corrections to the Kalman filter measurement stream. Further delays are introduced by the time it takes to process the Kalman filter measurement update and communicate the correction. Figure 3.9 illustrates this. As a result of these lags, one or more uncorrected measurement set may be processed by the Kalman filter, causing the closed-loop correction to be repeated. Overcorrection of a navigation system can cause instability with the navigation solution oscillating about the truth.

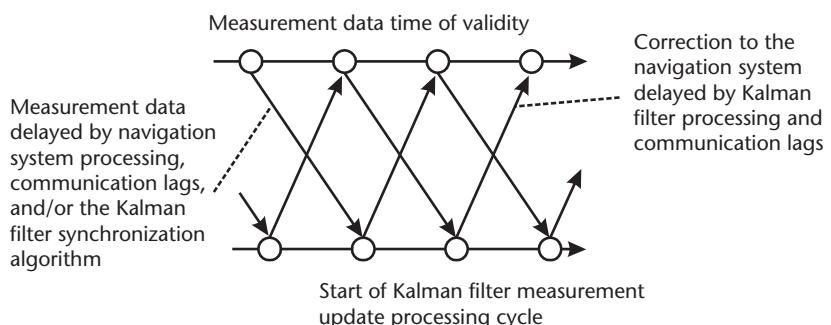


Figure 3.9 Processing lag in a closed-loop Kalman filter.

The optimal solution to this problem is to apply corrections to the measurement innovations or the data store in Figure 3.8. However, a simpler solution is to down-weight the Kalman gain, K , either directly or via the measurement noise covariance, R .

3.3.5 Kalman Filter Design Process

A good design philosophy [12] for a Kalman filter is to first select as states all known errors or properties of the system that are modelable, observable, and contribute to the desired output of the overall system, generally a navigation solution. System and measurement models should then be derived based on this state selection.

A software simulation should be developed, containing a version of the Kalman filter in which groups of states may be deselected and different phases of the algorithm run at different rates. With all states selected and all Kalman filter phases run at the fastest rate, the filter should be tuned and a covariance analysis performed to check that it meets the requirements. Processor load need not be a major consideration at this stage.

Assuming the requirements are met, simulation runs should then be conducted with different groups of Kalman filter states deselected and their effects modeled as system noise. Runs should also be conducted with phases of the Kalman filter run at a range of slower rates. Combinations of these configurations should also be investigated. Those changes that have least effect on Kalman filter performance for a given reduction in processor load should then be implemented in turn until the computational load falls within the available processing capacity.

The reduced Kalman filter should then be carefully retuned and assessed by simulation and trials to verify its performance.

3.4 Extensions to the Kalman Filter

The derivation of the Kalman filter algorithm is based on a number of assumptions about the properties of the states estimated and noise sources accounted for. However, these assumptions do not always apply to real navigation systems. This section looks at how the basic Kalman filter technique may be extended to handle a nonlinear measurement or system model, time-correlated noise, unknown system or measurement noise standard deviations, and non-Gaussian measurement distributions. In addition, Kalman smoothing techniques, which take advantage of the extra information available in postprocessed applications, are discussed.

3.4.1 Extended and Linearized Kalman Filter

In a standard Kalman filter, the measurement model is assumed to be linear (i.e., the measurement vector, \mathbf{z} , is a linear function of the state vector, \mathbf{x}). This is not always the case for real systems. In some applications, such as most INS alignment and calibration problems, a linear approximation of the measurement model is useful, though this can introduce small errors. However, for applications processing

ranging measurements, such as a GNSS navigation filter, the measurement model is highly nonlinear.³

The system model is also assumed to be linear in the standard Kalman filter (i.e., $\dot{\mathbf{x}}$ is a linear function of \mathbf{x}). Closed-loop correction of the system using the state estimates (Section 3.2.6) can often be used to maintain a linear approximation in the system model. However, it is not always possible to perform the necessary feedback to the system. An example of this is total-state INS/GNSS integration (see Section 12.1.1), where the absolute position, velocity, and attitude are estimated rather than the errors therein.

A nonlinear version of the Kalman filter is the *extended Kalman filter* (EKF). In an EKF, the system matrix, \mathbf{F} , and measurement matrix, \mathbf{H} , can be replaced in the state propagation and update equations by nonlinear functions of the state vector, respectively, $\mathbf{f}(\mathbf{x})$ and $\mathbf{h}(\mathbf{x})$. It is common in navigation applications to combine the measurement-update phase of the EKF with the system-propagation phase of the standard Kalman filter. The reverse combination may also be used, though it is rare in navigation.

The system dynamic model of the EKF is

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \mathbf{G}(t) \mathbf{w}_s(t) \quad (3.48)$$

where the nonlinear function of the state vector, \mathbf{f} , replaces the product of the system matrix, and state vector and the other terms are as defined in Section 3.2.4. The state vector propagation equation is thus

$$\hat{\mathbf{x}}_k^- = \hat{\mathbf{x}}_{k-1}^+ + \int_{t-\tau_s}^t \mathbf{f}(\hat{\mathbf{x}}, t') dt' \quad (3.49)$$

replacing (3.10).

In the EKF, it is assumed that the error in the state vector estimate is much smaller than the state vector, enabling a linear system model to be applied to the state vector residual:

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}(t) \delta\mathbf{x}(t) + \mathbf{G}(t) \mathbf{w}_s(t) \quad (3.50)$$

The conventional error covariance propagation equation, (3.11), may thus be used with the transition matrix linearized about the state vector estimate using (3.24):

$$\Phi_{k-1} \approx \exp(\mathbf{F}_{k-1} \tau_s)$$

solved using a power-series expansion as in the conventional Kalman filter, but where

$$\mathbf{F}_{k-1} = \left. \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x} = \hat{\mathbf{x}}_{k-1}^+} \quad (3.51)$$

The measurement model of the EKF is

$$\mathbf{z}(t) = \mathbf{h}(\mathbf{x}(t)) + \mathbf{w}_m(t) \quad (3.52)$$

where \mathbf{h} is a nonlinear function of the state vector. The state vector is then updated with the true measurement vector using

$$\begin{aligned} \hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k [\mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-)] \\ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k \delta \mathbf{z}_k^- \end{aligned} \quad (3.53)$$

replacing (3.16), where from (3.7) and (3.52), the measurement innovation is

$$\begin{aligned} \delta \mathbf{z}_k^- &= \mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-) \\ &= \mathbf{h}(\mathbf{x}_k) - \mathbf{h}(\hat{\mathbf{x}}_k^-) + \mathbf{w}_{mk} \end{aligned} \quad (3.54)$$

Once the state vector estimate has converged with its true counterpart, the measurement innovations will be small, so they can legitimately be modeled as a linear function of the state vector where the full measurements cannot. Thus,

$$\delta \mathbf{z}_k^- = \mathbf{H}_k \delta \mathbf{x}_k^- + \mathbf{w}_{mk} \quad (3.55)$$

where

$$\mathbf{H}_k = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x} = \hat{\mathbf{x}}_k^-} = \frac{\partial \mathbf{z}(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x} = \hat{\mathbf{x}}_k^-} \quad (3.56)$$

A consequence of this linearization of \mathbf{F} and \mathbf{H} is that the error covariance matrix, \mathbf{P} , and Kalman gain, \mathbf{K} , are functions of the state estimates. This can occasionally cause stability problems, and the EKF is more sensitive to the tuning of the \mathbf{P} -matrix initialization than a standard Kalman filter.³

An alternative that maintains an error covariance and Kalman gain that are independent of the state estimates is the linearized Kalman filter. This takes the same form as the EKF with the exception that the system and measurement matrices are linearized about a predetermined state vector, \mathbf{x}^P :

$$\mathbf{F}_{k-1} = \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x} = \mathbf{x}_{k-1}^P}, \quad \mathbf{H}_k = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x} = \mathbf{x}_k^P} \quad (3.57)$$

A suitable application is guided weapons, where the approximate trajectory is known prior to launch and the Kalman filter is estimating the navigation solution.

There are also higher order nonlinear filtering algorithms which do not linearize the error covariance propagation and update [2].³ These include the unscented Kalman filter (UKF), also known as the sigma-point Kalman filter [13]. The UKF uses the initial error covariance, \mathbf{P}_{k-1}^+ , system noise covariance, \mathbf{Q}_{k-1} , and measurement noise covariance, \mathbf{R}_k , matrices to generate a set of parallel state vectors.

These undergo a nonlinear system propagation and measurement update and are then used to generate the updated error covariance matrix, \mathbf{P}_k^+ .

3.4.2 Time-Correlated Noise and the Schmidt-Kalman Filter

In Kalman filtering, it is assumed that all measurement errors, \mathbf{w}_m , are time uncorrelated; in other words, the measurement noise is white. In practice this is often not the case. For example, Kalman filters in navigation often input measurements output by another Kalman filter, a loop filter, or another estimation algorithm. There may also be time-correlated variation in the lever arm between navigation systems. A Kalman filter attributes the time-correlated parts of the measurement innovations to the states. Consequently, correlated measurement noise can potentially corrupt the state estimates.

There are three main ways to account for time-correlated measurement noise in a Kalman filter. The optimal solution is to estimate the time-correlated noise as additional Kalman filter states. However, this may not be practical due to observability or processing capacity limitations. The second, and simplest, option is to reduce the gain of the Kalman filter. The measurement update interval may be increased to match the measurement noise correlation time; the assumed measurement noise covariance, \mathbf{R} , may be increased; or the Kalman gain, \mathbf{K} , down-weighted. Measurement averaging may be used in conjunction with an increased update interval, provided the averaged measurement is treated as a single measurement for statistical purposes. These gain-reduction techniques will all increase the time it takes the Kalman filter to converge and the uncertainty of the estimates at convergence. The third method of handling time-correlated noise is to use a Schmidt-Kalman filter, which is described later in this section.

Another assumption of Kalman filters is that the system noise, \mathbf{w}_s , is not time correlated. However, the system often exhibits significant systematic and other time-correlated errors that are not estimated as states due to observability or processing power limitations, but that affect the states that are estimated. These errors must be accounted for.¹

Where the correlation times are relatively short, these system errors may be modeled as white noise. However, the white noise must overbound the correlated noise, affecting the Kalman filter's convergence properties. For error sources correlated over more than a minute or so, a white noise approximation does not effectively model how the effects of these error sources propagate with time. The solution is to use a Schmidt-Kalman filter with uncertain parameters [14]. This effectively increases the error covariance matrix, \mathbf{P} , to model the time-correlated noise. It can also account for time-correlated measurement noise.²

The covariance matrix of the unestimated parameters that account for time-correlated system and measurement noise is \mathbf{W} . These parameters are sometimes known as consider states. The correlation matrix, \mathbf{U} , models the correlation between the unestimated parameters and the states. The transition matrix and system noise covariance matrix for the correlated noise parameters are, respectively, Φ_U and \mathbf{Q}_U . An additional transition matrix, Ψ , models how the states vary with the unestimated parameters. Consequently, the basic error covariance system propagation equation, (3.11), is replaced by

$$\begin{pmatrix} \mathbf{P}_k^- & \mathbf{U}_k^- \\ \mathbf{U}_k^{-T} & \mathbf{W}_k^- \end{pmatrix} = \begin{pmatrix} \Phi_{k-1} & \Psi_{k-1} \\ 0 & \Phi_{U,k-1} \end{pmatrix} \begin{pmatrix} \mathbf{P}_{k-1}^+ & \mathbf{U}_{k-1}^+ \\ \mathbf{U}_{k-1}^{+T} & \mathbf{W}_{k-1}^+ \end{pmatrix} \begin{pmatrix} \Phi_{k-1}^T & 0 \\ \Psi_{k-1}^T & \Phi_{U,k-1}^T \end{pmatrix} + \begin{pmatrix} \mathbf{Q}_{k-1} & 0 \\ 0 & \mathbf{Q}_{U,k-1} \end{pmatrix} \quad (3.58)$$

This separates into different propagation equations for the error covariance, correlated noise covariance, and correlation matrices, noting that the state propagation equation, (3.10), is unchanged from the standard Kalman filter:

$$\begin{aligned} \mathbf{P}_k^- = & \Phi_{k-1} \mathbf{P}_{k-1}^+ \Phi_{k-1}^T + \Phi_{k-1} \mathbf{U}_{k-1}^+ \Psi_{k-1}^T + \Psi_{k-1} \mathbf{U}_{k-1}^{+T} \Phi_{k-1}^T \\ & + \Psi_{k-1} \mathbf{W}_{k-1}^+ \Psi_{k-1}^T + \mathbf{Q}_{k-1} \end{aligned} \quad (3.59)$$

$$\mathbf{W}_k^- = \Phi_{U,k-1} \mathbf{W}_{k-1}^+ \Phi_{U,k-1}^T + \mathbf{Q}_{U,k-1} \quad (3.60)$$

$$\mathbf{U}_k^- = \Phi_{k-1} \mathbf{U}_{k-1}^+ \Phi_{U,k-1}^T + \Psi_{k-1} \mathbf{W}_{k-1}^+ \Phi_{U,k-1}^T \quad (3.61)$$

Defining \mathbf{J} as the matrix coupling the unestimated parameters to the measurement vector, corresponding to the \mathbf{H} matrix for the estimated states, the Kalman gain becomes

$$\mathbf{K}_k = (\mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{U}_k^- \mathbf{J}_k^T) (\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{H}_k \mathbf{U}_k^- \mathbf{J}_k^T + \mathbf{J}_k \mathbf{U}_k^{-T} \mathbf{H}_k^T + \mathbf{J}_k \mathbf{W}_k^- \mathbf{J}_k^T + \mathbf{R}_k)^{-1} \quad (3.62)$$

Apart from the redefinition of the Kalman gain, the state update equation, (3.16), is unchanged. The error covariance, correlated noise, and correlation matrices update as follows:

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^- - \mathbf{K}_k \mathbf{J}_k \mathbf{U}_k^{-T} \quad (3.63)$$

$$\mathbf{W}_k^+ = \mathbf{W}_k^- \quad (3.64)$$

$$\mathbf{U}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{U}_k^- - \mathbf{K}_k \mathbf{J}_k \mathbf{W}_k^- \quad (3.65)$$

Where none of the unestimated parameters are directly correlated with the measurements, there is only time-correlated system noise and $\mathbf{J} = 0$. Similarly, if none of the states vary as a function of the unestimated parameters, there is only time correlated measurement noise and $\Psi = 0$. Note that an unestimated parameter can impact both the system and measurement models in the same manner that a state can.

As the system propagation and measurement update of the error covariance matrix dominates the Kalman filter's processor load, the Schmidt-Kalman filter

typically uses at least a quarter as much processing power as a Kalman filter that estimates the time-correlated noise as states. This goes up to at least half the Kalman filter's load where all of the parameters are correlated with the measurements. The processor load can be reduced by neglecting parts of the correlation matrix, \mathbf{U} . However, great caution must be taken to avoid filter instability.

3.4.3 Adaptive Kalman Filter

For most applications, the Kalman filter's system noise covariance matrix, \mathbf{Q} , and measurement noise covariance matrix, \mathbf{R} , are determined during the development phase by laboratory measurements of the system, simulation, and trials. However, there are some cases where this cannot be done. For example, if an INS/GNSS integration algorithm or INS calibration algorithm is designed for use with a range of different inertial sensors, the system noise covariance will not be known in advance of operation. Similarly, if a transfer alignment algorithm (Section 13.1) is designed for use on different aircraft and weapon stores without prior knowledge of the flexure and vibration environment, the measurement noise covariance will not be known in advance. In other cases, the optimum Kalman filter tuning might vary with time. For example, the accuracy of GNSS measurements varies with the signal-to-noise level, satellite geometry, and multipath environment. For these applications, an adaptive Kalman filter may be used that estimates \mathbf{R} and/or \mathbf{Q} as it operates.³ There are two main approaches, innovation-based adaptive estimation (IAE) [15, 16] and multiple-model adaptive estimation (MMAE) [17].

The IAE method calculates either the system noise covariance, \mathbf{Q} ; the measurement noise covariance, \mathbf{R} ; or both from the measurement innovation statistics. The first step is the calculation of the covariance of the last n measurement innovations, \mathbf{C} :

$$\tilde{\mathbf{C}}_{\delta \mathbf{z}, k}^{-} = \frac{1}{n} \sum_{j=k-n}^{k} \delta \mathbf{z}_j^{-} \delta \mathbf{z}_j^{-T} \quad (3.66)$$

This can be used to compute \mathbf{Q} and/or \mathbf{R} :

$$\begin{aligned} \tilde{\mathbf{Q}}_k &= \mathbf{K}_k \tilde{\mathbf{C}}_{\delta \mathbf{z}, k}^{-} \mathbf{K}_k^T \\ \tilde{\mathbf{R}}_k &= \tilde{\mathbf{C}}_{\delta \mathbf{z}, k}^{-} - \mathbf{H}_k \mathbf{P}_k^{-} \mathbf{H}_k^T \end{aligned} \quad (3.67)$$

Initial values of \mathbf{Q} and \mathbf{R} must be provided for use while the first set of measurement innovation statistics is compiled. These values should be selected as cautiously as possible.³

The MMAE method uses a bank of parallel Kalman filters with different values of the system and/or measurement noise covariance matrices, \mathbf{Q} and \mathbf{R} . Different initial values of the error covariance matrix, \mathbf{P} , may also be used. Each of the Kalman filter hypotheses, denoted by the index i , is allocated a probability as follows [3, 4]:

$$p_{k,i} = \frac{p'_{k,i}}{\sum_{j=1}^l p'_{k,j}} \quad (3.68)$$

$$p'_{k,i} = \frac{p_{k-1,i}}{\sqrt{(2\pi)^m |\mathbf{H}_{k,i} \mathbf{P}_k^- \mathbf{H}_{k,i}^\top + \mathbf{R}_{k,i}|}} \exp \left[-\frac{1}{2} \delta \mathbf{z}_{k,i}^\top (\mathbf{H}_{k,i} \mathbf{P}_k^- \mathbf{H}_{k,i}^\top + \mathbf{R}_{k,i})^{-1} \delta \mathbf{z}_{k,i} \right]$$

where m is the number of components of the measurement vector and l is the number of filter hypotheses. Note that the matrix inversion is already performed as part of the Kalman gain calculation. The filter hypothesis with the smallest normalized measurement innovations is most consistent with the measurement stream, so is allocated the largest probability.

Over time, the probability of the best filter hypothesis will approach unity, while the others approach zero. To make best use of the available processing capacity, weak hypotheses should be deleted and the strongest hypothesis periodically subdivided to refine the filter tuning and allow it to respond to changes in the system.

The overall state vector estimate and error covariance are obtained as follows:

$$\hat{\mathbf{x}}_k^+ = \sum_{i=1}^l p_{k,i} \hat{\mathbf{x}}_{k,i}^+ \quad (3.69)$$

$$\mathbf{P}_k^+ = \sum_{i=1}^l p_{k,i} [\mathbf{P}_{k,i}^+ + (\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_k^+) (\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_k^+)^T] \quad (3.70)$$

noting that the error covariance matrix must account for the spread in the state vector estimates of the filter hypotheses as well as the error covariance of each hypothesis.

Comparing the IAE and MMAE adaptive Kalman filter techniques, the latter is more computationally intensive, as a bank of Kalman filters must be processed instead of just one. However, in an IAE Kalman filter, the system noise covariance, measurement noise covariance, error covariance, and Kalman gain matrices may all be functions of the state estimates, whereas they are independent in the MMAE filter bank (assuming conventional Kalman filters rather than EKFs). Consequently, the MMAE is less prone to filter instability.

3.4.4 Multiple-Hypothesis Filtering

An assumption of the standard Kalman filter is that the measurements have Gaussian distributions, enabling the measurement vector to be modeled as a mean, \mathbf{z} , and covariance, \mathbf{R} . However, this is not the case for every navigation system. Ranging systems can produce bimodal position measurements where there are insufficient signals for a unique fix, while some feature-matching techniques (Chapter 11) can produce a fix in the form of a highly irregular position distribution. To process these measurements in a Kalman filter-based estimation algorithm, they must first be expressed as a sum of Gaussian distributions, known as hypothe-

ses, each with a mean, \mathbf{z}_i , a covariance, \mathbf{R}_i , and also a probability, p_i . A probability score, p_0 , should also be allocated to the null hypothesis, representing the probability that none of the other hypotheses are correct. The probability scores sum to unity:

$$\sum_{i=0}^{n_k} p_{k,i} = 1 \quad (3.71)$$

where n_k is the number of hypotheses and k denotes the Kalman filter iteration as usual.

There are three main methods of handling multiple-hypothesis measurements using Kalman filter techniques: best fix, weighted fix, and multiple-hypothesis filtering. The best-fix method is a standard Kalman filter that accepts the measurement hypothesis with the highest probability score and rejects the others. It should incorporate a prefiltering algorithm that rejects all of the measurement hypotheses where none is dominant. This method has the advantage of simplicity and can be effective where one hypothesis is clearly dominant on most iterations.

Weighted-fix techniques input all of the measurement hypotheses, weighted according to their probabilities, but maintain a single set of state estimates. An example is the probabilistic data association filter (PDAF) [18, 19], which is predominantly applied to target tracking problems. The system-propagation phase of the PDAF is the same as for a standard Kalman filter. In the measurement-update phase, the Kalman gain calculation is performed for each of the measurement hypotheses:

$$\mathbf{K}_{k,i} = \mathbf{P}_k^- \mathbf{H}_k^T [\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_{k,i}]^{-1} \quad (3.72)$$

The state vector and error covariance matrix are then updated using

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \sum_{i=1}^{n_k} p_{k,i} \mathbf{K}_{k,i} (\mathbf{z}_{k,i} - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \quad (3.73)$$

$$= \hat{\mathbf{x}}_k^- + \sum_{i=1}^{n_k} p_{k,i} \mathbf{K}_{k,i} \delta \mathbf{z}_{k,i}^-$$

$$\mathbf{P}_k^+ = \left[\mathbf{I} - \left(\sum_{i=1}^{n_k} p_{k,i} \mathbf{K}_{k,i} \right) \mathbf{H}_k \right] \mathbf{P}_k^- + \sum_{i=1}^{n_k} p_{k,i} (\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_k^+) (\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_k^+)^T \quad (3.74)$$

where

$$\begin{aligned} \hat{\mathbf{x}}_{k,i}^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_{k,i} (\mathbf{z}_{k,i} - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \\ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_{k,i} \delta \mathbf{z}_{k,i}^- \end{aligned} \quad (3.75)$$

Note that, where the measurement hypotheses are widely spread compared to the prior state uncertainties, the state uncertainty (root diagonals of \mathbf{P}) can be larger following the measurement update; this cannot happen in a standard Kalman filter.

Compared to the best-fix technique, the PDAF has the advantage that it incorporates all true measurement hypotheses, but the disadvantage that it also incorporates all of the false hypotheses. It is most suited to applications where false hypotheses are not correlated over successive iterations or the truth is a combination of overlapping Gaussian measurement hypotheses.

Where false measurement hypotheses are time correlated, a multiple-hypothesis Kalman filter (MHKF) enables multiple state vector hypotheses to be maintained in parallel using a bank of Kalman filters. The technique was originally developed for target tracking [20], so is often known as multiple-hypothesis tracking (MHT). As the true hypothesis is identified over a series of filter cycles, the false measurement hypotheses are gradually eliminated from the filter bank. Like the MMAE filter, the MHKF maintains a set of state vector and error covariance matrix hypotheses that are propagated independently through the system model using the conventional Kalman filter equations. Each of these hypotheses has an associated probability score.

For the measurement update phase, the filter bank is split into $(n_k + 1)l$ hypotheses, combining each state vector hypothesis with each measurement hypothesis and the null measurement hypothesis. Figure 3.10 shows the principle. A conventional Kalman filter update is then performed for each hypothesis and a probability score allocated that multiplies the probabilities of the state and measurement hypotheses. The new hypothesis must also be scored for consistency between the state vector and measurement hypotheses; a probability weighting similar to that used for the MMAE [see (3.68)] is suitable. Following this, the probability scores must be renormalized, noting that the scores for the null measurement hypotheses should remain unchanged.

It is clearly impractical for the number of state vector hypotheses to increase on each iteration of the Kalman filter, so the measurement update process must conclude with a reduction in the number of hypotheses to l . This is done by merging hypotheses. The exact approach varies between implementations, but, generally, similar hypotheses are merged with each other and the weakest hypotheses, in terms of their probability scores, are merged into their nearest neighbor. Hypotheses with probability scores below a certain minimum may simply be deleted. A pair of hypotheses, denoted by indices α and β , are merged into a new hypothesis, denoted by γ using

$$p_{k,\gamma} = p_{k,\alpha} + p_{k,\beta} \quad (3.76)$$

$$\hat{\mathbf{x}}_{k,\gamma}^+ = \frac{p_{k,\alpha} \hat{\mathbf{x}}_{k,\alpha}^+ + p_{k,\beta} \hat{\mathbf{x}}_{k,\beta}^+}{p_{k,\gamma}} \quad (3.77)$$

$$\mathbf{P}_{k,\gamma}^+ = \sum_{i=\alpha, \beta} p_{k,i} [\mathbf{P}_{k,i}^+ + (\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_{k,\gamma}^+) (\hat{\mathbf{x}}_{k,i}^+ - \hat{\mathbf{x}}_{k,\gamma}^+)^T] \quad (3.78)$$

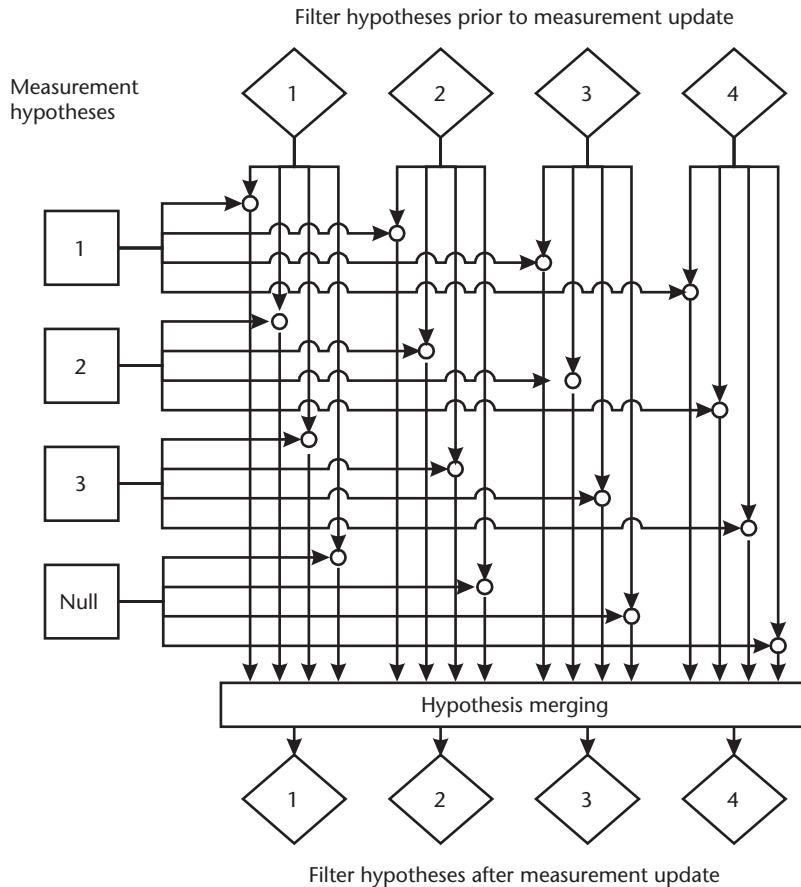


Figure 3.10 Multiple-hypothesis Kalman filter measurement update ($I = 4$, $n_k = 3$).

The overall state vector estimate and error covariance can either be the weighted average of all the hypotheses, obtained using (3.69) and (3.70), or the highest probability hypothesis, depending on the needs of the application. Where closed-loop correction (Section 3.2.6) is used, it is not possible to feed back corrections from the individual filter hypotheses, as this would be contradictory; the closed-loop feedback must come from the filter bank as a whole. The corrections fed back to the system must also be subtracted from all of the state vector hypotheses in order to maintain constant differences between the hypotheses. Thus the state estimates are not zeroed at feedback, so state vector propagation using (3.10) must take place in the same manner as for the open-loop Kalman filter.

The iterative Gaussian mixture approximation of the posterior (IGMAP) method [21], which can operate with either a single or multiple hypothesis state vector, combines the fitting of a set of Gaussian distributions to the measurement probability distribution and the measurement-update phase of the estimation algorithm into a single iterative process. By moving the approximation as a sum of Gaussian distributions from the beginning to the end of the measurement-update cycle, the residuals of the approximation process are reduced, producing more accurate state estimates. The system-propagation phase of IGMAP is the same as

for a conventional Kalman filter or MHKF. However, IGMAP does require more processing capacity than a PDAF or MHKF.

The need to apply a Gaussian approximation to the measurement noise and system noise distributions can be removed altogether by using a Monte-Carlo estimation algorithm, such as a particle filter [22–24] or Metropolis-coupled Monte-Carlo Markov chain (MCMCMC) [25]. These also avoid the need for linear system and measurement models. However, Monte-Carlo methods require at least an order of magnitude more processing power than Kalman filter-based estimation algorithms.

3.4.5 Kalman Smoothing

The Kalman filter is designed for real-time applications. It estimates the properties of a system at a given time using measurements of the system up to that time. However, for applications such as surveillance and testing, where the properties of a system are required after the event, a Kalman filter effectively throws away half the measurement data as it does not use measurements taken after the time of interest.³

The Kalman smoother is the extension of the Kalman filter using measurement information after as well as before the time of interest. This leads to more accurate state estimates for nonreal-time applications. There are two main methods, the forward-backward filter [2, 26] and the Rauch, Tung, and Striebel (RTS) method [4, 27].

The forward-backward filter comprises two Kalman filters, a forward filter and a backward filter. The forward filter is a standard Kalman filter. The backward filter is a Kalman filter algorithm working backward in time from the end of the data segment to the beginning. The two filters are treated as independent, so the backward filter must not be initialized with the final solution of the forward filter. The smoothed estimates are obtained simply by combining the estimates of the two filters, weighted according to the ratio of their error covariance matrices:¹

$$\begin{aligned}\hat{\mathbf{x}}_k^+ &= (\mathbf{P}_{f,k}^+ + \mathbf{P}_{b,k}^+)^{-1} (\mathbf{P}_{f,k}^+ \hat{\mathbf{x}}_{f,k}^+ + \mathbf{P}_{b,k}^+ \hat{\mathbf{x}}_{b,k}^+) \\ \mathbf{P}_k^+ &= (\mathbf{P}_{f,k}^{+1} + \mathbf{P}_{b,k}^{+1})^{-1}\end{aligned}\quad (3.79)$$

where the subscripts f and b refer to the forward and backward filters, respectively.² The index, k , refers to the same point in time for both filters, so the backward filter must count backward. Figure 3.11 shows how the state uncertainty varies with time for the forward, backward and combined filters. It is only necessary to store the state vectors and error covariance matrices and perform the matrix inversion at the points of interest. Note that it is not necessary to run the forward filter beyond the last point of interest and the backward filter beyond the first point of interest.

In the RTS method, a conventional Kalman filter runs forward in time, but stores the state vector, \mathbf{x} , and the error covariance matrix, \mathbf{P} , after each system propagation and measurement update. The transition matrix, Φ , is also stored. Once the end of the data set is reached, smoothing begins, starting at the end and

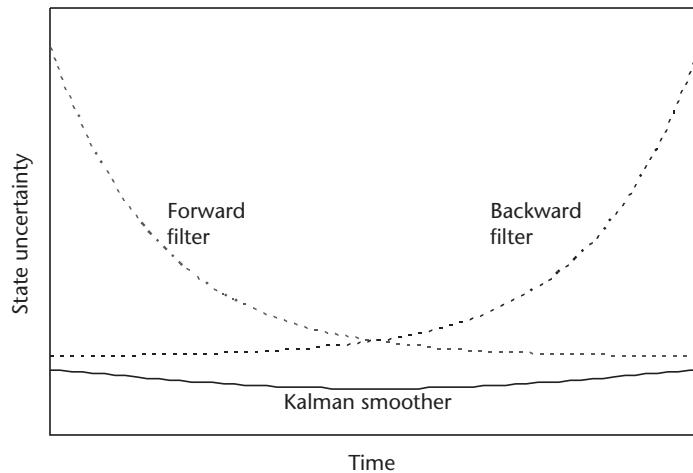


Figure 3.11 Forward-backward Kalman smoother state uncertainty. (From: [reference number??]
© 2002 QinetiQ Ltd. Reprinted with permission.)

working back to the beginning. The smoothing gain on each iteration, A_k , is given by

$$A_k = P_k^+ \Phi_k^T (P_{k+1}^-)^{-1} \quad (3.80)$$

The smoothed state vector, \hat{x}_k^s , and error covariance, P_k^s , are then given by

$$\begin{aligned} \hat{x}_k^s &= \hat{x}_k^+ + A_k (\hat{x}_{k+1}^s - \hat{x}_{k+1}^-) \\ P_k^s &= P_k^+ + A_k (P_{k+1}^s - P_{k+1}^-) A_k^T \end{aligned} \quad (3.81)$$

Where the smoothed solution is required at all points, the RTS method is more efficient, whereas the forward-backward method is more efficient where a smoothed solution is only required at a single point.

Kalman smoothing can also be used to provide a quasi-real-time solution by making use of information from a limited period after the time of interest. A continuous solution is then output at a fixed lag. This can be useful for tracking applications, such as logistics, security, and road-user charging, that require bridging of GNSS outages.

References

- [1] Jazwinski, A. H., *Stochastic Processes and Filtering Theory*, San Diego, CA: Academic Press, 1970.
- [2] Gelb, A., (ed.), *Applied Optimal Estimation*, Cambridge, MA: MIT Press, 1974.
- [3] Maybeck, P. S., *Stochastic Models, Estimation and Control*, Vols. 1–3, New York: Academic Press, 1979.
- [4] Brown, R. G., and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, 3rd ed., New York: Wiley, 1997.

- [5] Grewal, M. S., and A. P. Andrews, *Kalman Filtering: Theory and Practice*, 2nd ed., New York: Wiley, 2000.
- [6] Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *ASME Transactions, Series D: Journal of Basic Engineering*, Vol. 82, 1960, pp. 35–45.
- [7] Groves, P. D., *Principles of Integrated Navigation* (course notes), QinetiQ Ltd., 2002.
- [8] Golub, G. H., and C. F. Van Loan, *Matrix Computations*, Baltimore, MD: Johns Hopkins University Press, 1983.
- [9] Rogers, R. M., *Applied Mathematics in Integrated Navigation Systems*, Reston, VA: AIAA, 2000.
- [10] Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, New York: Wiley, 2001.
- [11] Bierman, G. L., *Factorization Methods for Discrete Sequential Estimation, Mathematics in Science and Engineering*, Vol. 128, New York: Academic Press, 1977.
- [12] Stimac, L. W., and T. A. Kennedy, "Sensor Alignment Kalman Filters for Inertial Stabilization Systems," *Proc. IEEE PLANS* Monterey, CA, March 1992, pp. 321–334.
- [13] Julier, S. J., and J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," *Proc. AeroSense: The 11th Int. Symp. on Aerospace/Defence Sensing, Simulation and Controls*, SPIE, 1997.
- [14] Schmidt, S. F., "Application of State Space Methods to Navigation Problems," in *Advanced in Control Systems*, Vol. 3, C. T. Leondes, (ed.), New York: Academic Press, 1966.
- [15] Mehra, R. K., "Approaches to Adaptive Filtering," *IEEE Trans. on Automatic Control*, Vol. AC-17, 1972, pp. 693–698.
- [16] Mohammed, A. H., and K. P. Schwarz, "Adaptive Kalman Filtering for INS/GPS," *Journal of Geodesy*, Vol. 73, 1999, pp. 193–203.
- [17] Magill, D. T., "Optimal Adaptive Estimation of Sampled Stochastic Processes," *IEEE Trans. on Automatic Control*, Vol. AC-10, 1965, pp. 434–439.
- [18] Bar-Shalom, Y., and T. E. Fortmann, *Tracking and Data Association*, New York: Academic Press, 1988.
- [19] Dezert, J., and Y. Bar-Shalom, "Joint Probabilistic Data Association for Autonomous Navigation," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 29, 1993, pp. 1275–1285.
- [20] Reid, D. B., "An Algorithm for Tracking Multiple Targets," *IEEE Trans. on Automatic Control*, Vol. AC-24, 1979, pp. 843–854.
- [21] Runnalls, A. R., P. D. Groves, and R. J. Handley, "Terrain-Referenced Navigation Using the IGMAP Data Fusion Algorithm," *Proc. ION 61st AM*, Boston, MA, June 2005, pp. 976–987.
- [22] Gordon, N. J., D. J. Salmond, and A. F. M. Smith, "A Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *Proc. IEE Radar Signal Process*, Vol. 140, 1993, pp. 107–113.
- [23] Gustafsson, F., et al., "Particle Filters for Positioning, Navigation and Tracking," *IEEE Trans. on Signal Processing*, Vol. 50, 2002, pp. 425–437.
- [24] Ristic, B., S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Norwood, MA: Artech House, 2004.
- [25] Geyer, C. J., and E. A. Thompson, "Constrained Monte-Carlo Maximum-Likelihood for Dependent Data," *Journal of the Royal Statistical Society Series B-Methodological*, Vol. 54, 1992, pp. 657–699.
- [26] Fraser, D. C., and J. E. Potter, "The Optimum Linear Smoother As a Combination of Two Optimum Linear Filters," *IEEE Trans. on Automatic Control*, Vol. 7, 1969, pp. 387–390.
- [27] Rauch, H. E., F. Tung, and C. T. Striebel, "Maximum Likelihood Estimates of Linear Dynamic Systems," *AIAA Journal*, Vol. 3, 1965, pp. 1445–1450.

Selected Bibliography

- Bar-Shalom, Y., X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*, New York: Wiley, 2001.
- Gustafsson, F., *Adaptive Filtering and Change Detection*, New York: Wiley, 2000.
- Minkler, G., and J. Minkler, *Theory and Applications of Kalman Filters*, Baltimore, MD: Magellan, 1993.
- Zarchan, P., and H. Musoff, *Fundamentals of Kalman Filtering: A Practical Approach*, Reston, VA: AIAA, 2000.

Endnotes

1. This and subsequent paragraphs are based on material written by the author for QinetiQ, so comprise QinetiQ copyright material.
2. End of QinetiQ copyright material.
3. This paragraph, up to this point, is based on material written by the author for QinetiQ, so comprises QinetiQ copyright material.

PART III

Navigation Systems

Inertial Sensors

Inertial sensors comprise accelerometers and gyroscopes, commonly abbreviated to gyros. An *accelerometer* measures specific force and a *gyroscope* measures angular rate, both without an external reference. Devices that measure the velocity, acceleration, or angular rate of a body with respect to features in the environment are not inertial sensors.

Most types of accelerometers measure specific force along a single sensitive axis. Similarly, most types of gyros measure angular rate about a single axis. An *inertial measurement unit* (IMU) combines multiple accelerometers and gyros, usually three of each, to produce a three-dimensional measurement of specific force and angular rate. An IMU is the sensor for an inertial navigation system, described in Chapter 5, which produces an independent three-dimensional navigation solution. New designs of INS all employ a strapdown architecture, whereby the inertial sensors are fixed with respect to the navigation system casing. The alternative platform architecture is discussed in Section 5.7. Lower grade IMUs are also used in AHRSSs, described in Section 10.1.4, and for PDR using step detection, discussed in Section 10.4. Inertial sensors also have many uses outside navigation as reviewed in [1].

This chapter describes the basic principles of accelerometer, gyro, and IMU technology, compares the different types of sensors, and reviews the error sources. Inertial sensor technology is reviewed in [1, 2].

Most accelerometers are either pendulous or use vibrating beams. Both technologies share the same basic principle and are described in Section 4.1. There are three main types of gyro technology: spinning mass, optical, and vibratory, each of which is based on a different physical principle. These are described in Section 4.2. The size, mass, performance, and cost of inertial sensors varies by several orders of magnitude, both within and between the different technologies. In general, higher performance sensors are larger and more massive as well as more costly.

Current inertial sensor development is focused on MEMS technology. This enables quartz and silicon sensors to be mass produced at low cost using etching techniques with several sensors on a single silicon wafer. MEMS sensors are small, light, and exhibit much greater shock tolerance than conventional mechanical designs. However, they currently offer relatively poor performance. Micro-optical-electro-mechanical systems (MOEMS) technology replaces the capacitive pick-off of many MEMS sensors with an optical readout, offering potential improvements in performance [3], but was still at the research stage at the time of writing.

The IMU regulates the power supplies to the inertial sensors, converts their outputs to engineering units, and transmits them on a data bus. It also calibrates

out many of the raw sensor errors. The IMU functions are discussed in Section 4.3, while Section 4.4 discusses the error behavior of the calibrated accelerometers and gyros.

There is no universally agreed definition of high-, medium-, and low-grade IMUs and inertial sensors. One author's medium grade can be another's high or low grade. Excluding the specialist technology employed in intercontinental ballistic missiles, IMUs, INSs, and inertial sensors may be grouped into five broad performance categories: marine, aviation, intermediate, tactical, and automotive.

The highest grades of inertial sensors discussed here are used in ships, submarines, and some spacecraft. A *marine-grade* INS can cost in excess of \$1 million or 1 million euros and offer a navigation-solution drift of less than 1.8 km in a day. Early systems offering that level of performance were very large, with a diameter of about a meter; current systems are much smaller.

Aviation-grade, or navigation-grade, INSs used in U.S. military aircraft are required to meet the standard navigation unit (SNU) 84 standard, specifying a maximum horizontal position drift of ~1.5 km in the first hour of operation. These INSs are also used in commercial airliners and in military aircraft worldwide. They cost around \$100,000 or 100,000 Euros and have a standard size of $178 \times 178 \times 249$ mm. An *intermediate-grade* IMU, about an order of magnitude poorer in performance terms, is used in small aircraft and helicopters and costs \$/€20,000–50,000.

A *tactical-grade* IMU can only be used to provide a useful stand-alone inertial navigation solution for a few minutes. However, an accurate long-term navigation solution can be obtained by integrating it with a positioning system, such as GPS. These systems typically cost between \$5,000 and \$20,000 or 5,000 and 20,000 Euros and are typically used in guided weapons and unmanned air vehicles (UAVs). Most are less than a liter in volume. Tactical grade covers a wide span of sensor performance, particularly for gyros.

The lowest grade of inertial sensors is often known as *automotive grade*. They tend to be sold as individual accelerometers and gyros, rather than as IMUs and are not accurate enough for inertial navigation, even when integrated with other navigation systems, but can be used in an AHRS or for PDR. They are typically used in pedometers, antilock braking systems (ABSs), active suspension, and airbags. Accelerometers cost around a dollar or euro, while gyro prices start at about \$/€10 [4]. Individual sensors are typically a few cubic centimeters in volume. Sometimes, the same MEMS inertial sensors are sold as automotive grade without calibration and tactical grade with calibration.

The range of inertial sensors from automotive to marine grade spans six orders of magnitude of gyro performance, but only three orders of magnitude of accelerometer performance. This is partly because gyro performance has more impact on navigation solution drift over periods in excess of about 40 minutes, as explained in Section 5.6.2.

4.1 Accelerometers

Figure 4.1 shows a simple accelerometer. A proof mass is free to move with respect to the accelerometer case along the accelerometer's sensitive axis, restrained by

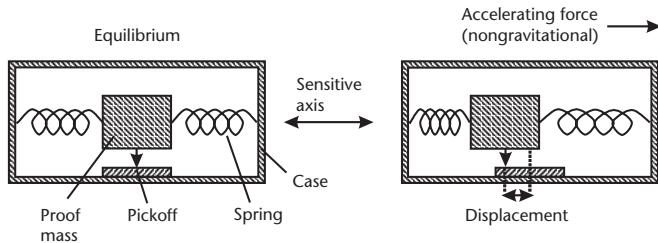


Figure 4.1 A simple accelerometer.

springs. A pickoff measures the position of the mass with respect to the case. When an accelerating force along the sensitive axis is applied to the case, the proof mass will initially continue at its previous velocity, so the case will move with respect to the mass, compressing one spring and stretching the other. Stretching and compressing the springs alters the forces they transmit to the proof mass from the case. Consequently, the case will move with respect to the mass until the acceleration of the mass due to the asymmetric forces exerted by the springs matches the acceleration of the case due to the externally applied force. The resultant position of the mass with respect to the case is proportional to the acceleration applied to the case. By measuring this with a pickoff, an acceleration measurement is obtained. The exception to this is acceleration due to the gravitational force. Gravitation acts on the proof mass directly, not via the springs, and applies the same acceleration to all components of the accelerometer, so there is no relative motion of the mass with respect to the case. Therefore, all accelerometers sense specific force, the nongravitational acceleration, not the total acceleration (see Section 2.3.5).

The object frame for accelerometer measurements is the accelerometer case, while the reference frame is inertial space, and measurements are resolved along the sensitive axes of the accelerometers. Thus, an IMU containing an accelerometer triad measures the specific force of the IMU body with respect to inertial space in body axes, the vector \mathbf{f}_{ib}^b .

The accelerometer shown in Figure 4.1 is incomplete. The proof mass needs to be supported in the axes perpendicular to the sensitive axis, and damping is needed to limit oscillation of the proof mass. However, all accelerometer designs are based on the basic principle shown. Practical accelerometers used in strapdown navigation systems follow either a pendulous or vibrating-beam design, both of which are discussed later. Pendulous designs have been around for decades, while vibrating-beam accelerometers originated in the 1980s. Both types of accelerometer may be built using either conventional mechanical construction or MEMS technology. MEMS accelerometers of either design may be built with sensitive axes, both in the plane of the device and perpendicular to that plane, enabling a three-axis accelerometer triad and associated electronics to be etched onto a single silicon chip [5]. A third type of accelerometer, the pendulous integrating gyro accelerometer (PIGA), is only suitable for use in a platform INS but can exhibit very high precision. In addition, research has been conducted into a number of novel accelerometer designs making use of optical, MEMS, and even atom interferometry techniques [1, 6].

4.1.1 Pendulous Accelerometers

Figure 4.2 shows a mechanical open-loop *pendulous* accelerometer. The proof mass is attached to the case via a pendulous arm and hinge, forming a pendulum. This leaves the proof mass free to move along the sensitive axis while supporting it in the other two axes. A pair of springs or a single spring is used to transmit force from the case to the pendulum along the sensitive axis while the hinge provides damping. Further damping may be obtained by filling the case with oil.

Although the open-loop design produces a practical accelerometer, its performance is severely limited by three factors. First, the resolution of the pick-off, typically a variable resistor, is relatively poor. Second, the force exerted by a spring is only approximately a linear function of its compression or extension, displaying hysteresis as well as nonlinearity. Finally, the sensitive axis is perpendicular to the pendulous arm so, as the pendulum moves, the sensitive axis moves with respect to the case. This results in both nonlinearity of response along the desired sensitive axis and sensitivity to orthogonal specific force.

To resolve these problems, precision accelerometers use a closed-loop, or *force-feedback*, configuration [1, 2]. In a force-feedback accelerometer, a torquer is used to maintain the pendulous arm at a constant position with respect to the case, regardless of the specific force to which the accelerometer is subject. The pickoff detects departures from the equilibrium position, and the torquer is adjusted to return the pendulum to that position. In a force-feedback accelerometer, the force exerted by the torquer, rather than the pickoff signal, is proportional to the applied specific force. Figure 4.3 depicts a mechanical force-feedback accelerometer. The torquer comprises an electromagnet mounted on the pendulum and a pair of permanent magnets of opposite polarity mounted on either side of the case. The diagram shows a capacitive pickoff, comprising four capacitor plates, mounted such that two capacitors are formed between the case and pendulum. As the pendulum moves, the capacitance of one pair of plates increases while that of the other decreases. Alternatively, an inductive or optical pickoff may be used.

The closed-loop configuration ensures that the sensitive axis remains aligned with the accelerometer case, while the torquer offers much greater dynamic range and linearity than the open-loop accelerometer's springs and pickoff. However, a drawback is that the pendulum is unrestrained when the accelerometer is unpowered, risking damage in transit, particularly where the case is gas-filled rather

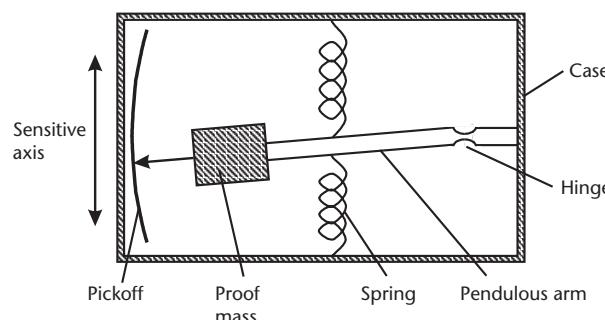


Figure 4.2 Mechanical open-loop pendulous accelerometer.

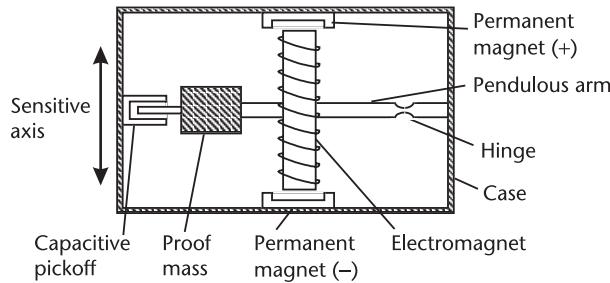


Figure 4.3 Mechanical force-feedback pendulous accelerometer. (After: [1].)

than oil-filled. The design of the hinge, pendulous arm, proof mass, torquer, pickoff system, and control electronics all affect performance. By varying the component quality, a range of different grades of performance can be offered at different prices.

Both open-loop and closed-loop pendulous MEMS accelerometers are available, with the latter using an electrostatic, rather than magnetic, torquer. The pickoff may be capacitive, as described earlier, or a resistive element mounted on the hinge, whose resistance varies as it is stretched and compressed.

4.1.2 Vibrating-Beam Accelerometers

The *vibrating-beam accelerometer* (VBA) or resonant accelerometer retains the proof mass and pendulous arm from the pendulous accelerometer. However, the proof mass is supported along the sensitive axis by a vibrating beam, largely constraining its motion with respect to the case. When a force is applied to the accelerometer case along the sensitive axis, the beam pushes or pulls the proof mass, causing the beam to be compressed in the former case and stretched in the latter. The beam is driven to vibrate at its resonant frequency by the accelerometer electronics. However, compressing the beam decreases the resonant frequency, whereas tensing it increases the frequency. Therefore, by measuring the resonant frequency, the specific force along the sensitive axis can be determined.

Performance is improved by using a pair of vibrating beams, arranged such that one is compressed while the other is stretched. They may support either a single proof mass or two separates masses; both arrangements are shown in Figure 4.4. Two-element *tuning fork* resonators are shown, as these are more balanced than single-element resonators. Larger scale VBAs all use quartz elements, as these provide a sharp resonance peak. MEMS VBAs have been fabricated out of both quartz and silicon.

The VBA is an inherently open-loop device. However, the proof mass is essentially fixed; there is no variation in the sensitive axis with respect to the casing.

4.2 Gyroscopes

This section describes the principles of the three main types of gyroscope: spinning mass, optical, and vibratory. The definition of a gyroscope was originally restricted

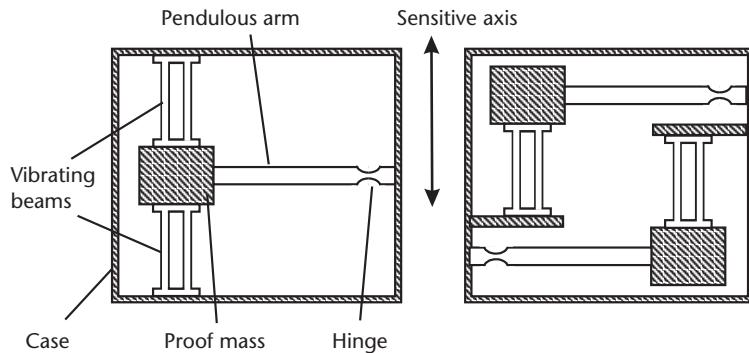


Figure 4.4 Vibrating beam accelerometers.

to the spinning-mass type, but now encompasses all angular-rate sensors that do not require an external reference. Jean Bernard Léon Foucault is credited with inventing the spinning-mass gyro in 1852, while Charles Stark Draper led the development of high-performance gyros of this type in the 1950s [7].

There are two main types of optical gyro. The RLG originated in the 1960s [8] as a high-performance technology, while the IFOG was developed in the 1970s [9] as a lower cost solution. Now, the performance ranges of the RLG and IFOG overlap. A resonant fiber-optic gyro (RFOG) and a micro-optic gyro (MOG) have also been developed [2].

Practical vibratory gyros were developed from the 1980s. All MEMS gyros operate on the vibratory principle, but larger vibratory gyros are also available and the technique spans the full performance range.

A number of other gyroscope technologies, including nuclear magnetic resonance, fluoric sensors, angular accelerometers, and atom interferometry techniques, have also been researched [1, 6].

The object frame for gyro measurements is the gyro case, while the reference frame is inertial space, and measurements are resolved along the sensitive axis of the gyros. Thus, an IMU containing a gyro triad measures the angular rate of the IMU body with respect to inertial space in body axes, the vector ω_{ib}^b .

4.2.1 Spinning-Mass Gyroscopes

Spinning-mass gyros operate on the principle of conservation of angular momentum. Part of Newton's second law of dynamics, this states that the angular momentum of a body with respect to inertial space will remain unchanged unless acted upon by a torque (force \times distance). Therefore, if a spinning mass is mounted in an instrument case such that it is free to rotate about both of the axes perpendicular to its spin axis, it will remain aligned with respect to inertial space as the case is rotated. Pickoffs that measure the orientation of the spinning mass with respect to the case thus provide measurements of the instrument case's attitude about two axes. Such a device is known as a *gyrocompass* and is used as a stand-alone attitude sensor on many ships and aircraft. However, the resolution of attitude pickoffs limits the usefulness of gyrocompasses for inertial navigation.

For strapdown inertial navigation, a spinning-mass gyro must measure the angular rate about an axis that is fixed with respect to the instrument case. Figure 4.5 shows a disc spinning about the y -axis with angular momentum vector \mathbf{h} . A torque, $\boldsymbol{\tau}$, is applied about the orthogonal z -axis. Consider the elements of the disc furthest away from the z -axis at the top and bottom of the diagram. As a result of the disc's spinning, these elements have a velocity of $\pm v_s$ along the z -axis. However, integrating the torque about the z -axis also gives these elements a velocity of $\mp v_t$ along the y -axis, changing their trajectory. Extending this to every element of the disc causes it to rotate about the x -axis, mutually perpendicular to both the spin and torque axes. This motion about the x -axis is known as *precession*.

Consider a small precession through an angle $\omega_p \delta t$ where ω_p is the precession angular rate and δt is a time increment. The resulting change in the angular momentum vector, $\delta\mathbf{h}$, is proportional to the magnitude of the original angular momentum \mathbf{h} and the precession angle $\omega_p \delta t$, but is mutually perpendicular. Thus,

$$\delta\mathbf{h} = \boldsymbol{\omega}_p \wedge \mathbf{h} \delta t \quad (4.1)$$

The applied torque is equal to the rate of change of angular momentum, so

$$\boldsymbol{\tau} = \boldsymbol{\omega}_p \wedge \mathbf{h} \quad (4.2)$$

Applying this to a spinning-mass gyro, if the case is rotated about an axis perpendicular to the spin axis, a torque about the mutually perpendicular axis must be applied to the spinning mass to keep it aligned with the case.

Figure 4.6 shows an open-loop single-degree-of-freedom spinning-mass gyro. The spinning mass or rotor and its driving motor are mounted on a pivot about which the motor/rotor assembly is free to move. This axis is known as the output axis. The axis that is mutually perpendicular to the output and spin axes is known as the input axis, the axis about which the gyro measures rotation. The motor/rotor assembly cannot move about the input axis with respect to the case. When the gyro case is rotated about the input axis, it applies torque about that axis to

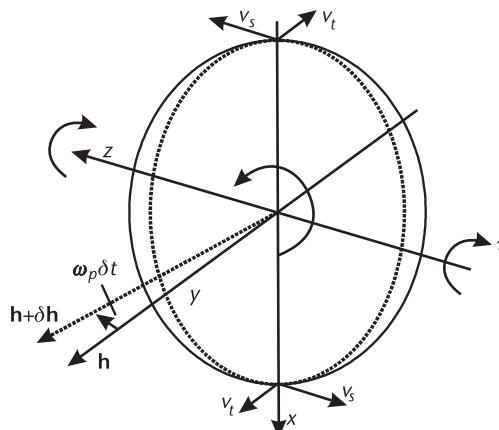


Figure 4.5 Effect of orthogonal torque on a spinning disc.

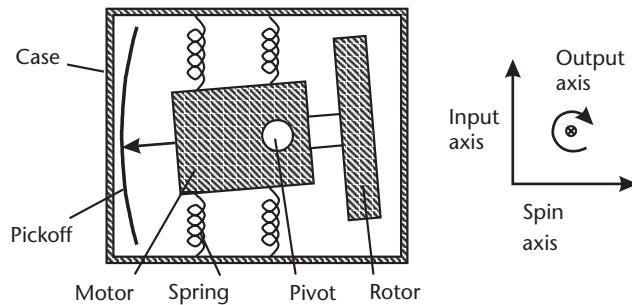


Figure 4.6 Open-loop single-degree-of-freedom spinning-mass gyro.

the rotor, causing rotation about the output axis. With no reaction to this, the motor/rotor assembly would precess until the spin and input axes are aligned. One or more springs are mounted between the case and the motor. These springs provide a balancing torque about the output axis, which precesses the rotor about the input axis, keeping it aligned with the casing. Alternatively, the output-axis pivot can be replaced by a torsion bar. Thus, the input and output axes act both as torquing and precession axes with the torques applied by the case about the input axis balanced by the torque applied by the springs about the output axis. The force applied by a spring is a function of its compression or tension. Consequently, when rotation is applied about the input axis, the motor/rotor assembly will rotate about the output axis until the torques balance. The orientation of the motor/rotor assembly with respect to the case is proportional to the angular rate about the input axis. Thus, a pickoff provides an angular rate measurement.

When the case is rotated about the output axis, the springs are compressed or stretched, changing the torque about this axis and stimulating precession about the input axis. The case then prevents the precession about the input axis by applying an opposing torque that precesses the motor/rotor assembly about the output axis such that it retains a constant orientation with respect to the casing. Hence, only rotation about the input axis produces a reading.

The open-loop spinning-mass gyro suffers from the same limitations as the open-loop pendulous accelerometer. The pickoff resolution is limited, the spring exhibits nonlinearity and hysteresis, and the orientation of the input axis with respect to the instrument casing varies. The solution is closed-loop operation. This replaces the springs with an electromagnetic torquer. A capacitive, inductive or optical pickoff is used to detect misalignment of the motor/rotor assembly with the gyro case and adjust the torquer to compensate. The torque exerted by the torquer is proportional to the angular rate of the gyro case about the input axis. Maximum sensitivity is obtained by maximizing the angular momentum of the spinning mass; this can be done by concentrating most of the rotor's mass around its edge. Different gyro performances can be offered at different prices by varying the quality of the spin motor, bearings, torquer, control electronics, and the size and symmetry of the rotor.

Spinning-mass gyros may also be designed to sense rotation about two orthogonal axes; these are known as two-degrees-of-freedom gyros. The simplest approach is to mount a single-degree-of-freedom gyro on a gimbal frame that is attached to

the gyro case by pivots on an orthogonal axis to the pivots supporting the rotor/motor assembly. An additional torquer and pickoff are then mounted between the gimbal frame and the gyro case. This is essentially one single-degree-of-freedom gyro inside another with the input and output axes of the “two gyros” exchanged. Other approaches include gyros with a floated spherical rotor and the dynamically tuned gyro, where the rotor is attached to the spin motor by a universal joint [1, 2]. For applications requiring angular-rate measurement about only two axes, a two-degrees-of-freedom gyro can be cheaper than two single-degree-of-freedom sensors.

4.2.2 Optical Gyroscopes

Optical gyroscopes work on the principle that, in a given medium, light travels at a constant speed in an inertial frame. If light is sent in both directions around a nonrotating closed-loop waveguide made of mirrors or optical fiber, the path length is the same for both beams. However, if the waveguide is rotated about an axis perpendicular to its plane, then, from the perspective of an inertial frame, the reflecting surfaces are moving further apart for light traveling in the same direction as the rotation and closer together for light traveling in the opposite direction. Thus, rotating the waveguide in the same direction as the light path increases the path length and rotating it in the opposite direction decreases the path length. This is known as the *Sagnac effect*. Figure 4.7 illustrates it. By measuring the changes in path length, the angular rate of the waveguide with respect to inertial space can be determined. Note that, from the perspective of the rotating frame, the path length remains unchanged but the speed of light changes.

4.2.2.1 Ring Laser Gyro

Figure 4.8 shows a ring laser gyro. A closed-loop tube with at least three arms is filled with a helium-neon gas mixture; this is known as a laser cavity. A high-reflectivity mirror is placed at each corner. Finally, a cathode and anode are used to apply a high potential difference across the gas, generating an electric field.

A gas atom can absorb energy from the electric field, producing an excited state of the atom. Excited states are unstable, so the atom will eventually return to its normal state, known as the ground state, by emitting the excess energy as a photon. There is some variation in the potential energies of the ground and excited

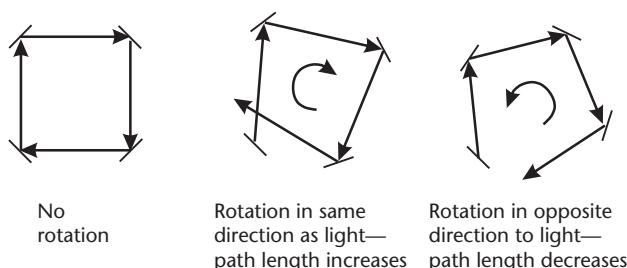


Figure 4.7 Effect of closed-loop waveguide rotation on path length.

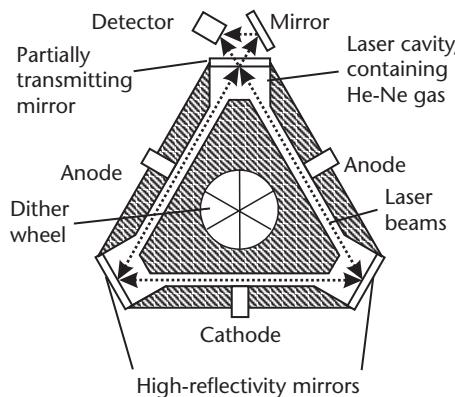


Figure 4.8 A typical ring laser gyro.

states, so the wavelengths of the spontaneously emitted photons are distributed over a resonance curve. The excited-state atoms can also be stimulated to emit photons by other photons in the laser cavity that are within the resonance curve. A photon produced by stimulated emission has the same wavelength, phase, and trajectory as the stimulating photon; this is known as coherence.

Photons of the same wavelength within the laser cavity interfere with each other. Where there are an integer number of wavelengths within the length of the laser cavity, the interference is constructive. This is known as a resonant mode. Otherwise, the interference is destructive. For a practical laser, the resonant modes of the cavity must have a narrower bandwidth than the resonance of the atom transition, and there should be more than one cavity mode within the atomic resonance curve. The laser will then adopt a lasing mode whereby the photons adopt the wavelength of the cavity mode closest to the atom resonance peak.

A ring laser has two lasing modes, one in each direction. If the laser cavity does not rotate, both modes have the same wavelength. However, if the laser cavity is rotated about an axis perpendicular to its plane, the cavity length is increased for the lasing mode in the direction of rotation and decreased for the mode in the opposite direction. Consequently, the lasing mode in the direction of rotation exhibits an increase in wavelength and decrease in frequency, while the converse happens for the other mode. In a ring laser gyro, one of the cavity mirrors is partially transmitting, enabling photons from both lasing modes to be focused on a detector, where they interfere. The beat frequency of the two modes is given by [1]

$$\Delta f \approx \frac{4A\omega_{\perp}}{\lambda_0} \quad (4.3)$$

where λ_0 is the wavelength of the nonrotating laser, A is the area enclosed by the RLG's light paths in the absence of rotation, and ω_{\perp} is the angular rate about an axis perpendicular to the plane of the laser cavity.

Because of scattering within the laser cavity, there is coupling between the clockwise and counterclockwise laser modes. At low angular rates, this prevents

the wavelengths of the two laser modes from diverging, a process known as lock-in. Thus, a basic ring laser gyro is unable to detect low angular rates. To mitigate this problem, most RLGs implement a dithering process, whereby the laser cavity is subject to low-amplitude, high-frequency angular vibrations about the sensitive axis with respect to the gyro case. Alternatively, the Kerr effect may be used to vary the refractive index within part of the cavity. This constantly changes the lock-in region in terms of the gyro case angular rate, which is the quantity to be measured [1, 2].

Most RLG triads contain three separate instruments. However, a few designs comprise a single laser cavity with lasing modes in three planes.

4.2.2.2 Interferometric Fiber-Optic Gyro

Figure 4.9 shows the main elements of an *interferometric fiber-optic gyro*, often abbreviated to just fiber-optic gyro (FOG) [1, 2, 10]. A broadband light source is divided using beam splitters into two equal portions that are then sent through a fiber-optic coil in opposite directions. The beam splitters combine the two beams at the detector, where the interference between them is observed. Two beam splitters, rather than one, are used so that both light paths include an equal number of transmissions and reflections. When the fiber-optic coil is rotated about an axis perpendicular to its plane, a phase change, ϕ_c , is introduced between the two light paths, given by

$$\phi_c \approx \frac{8\pi NA\omega_{\perp}}{\lambda_0 c_c} \quad (4.4)$$

where λ_0 is the wavelength of the light source, which does not change; A is the area enclosed by the coil; N is the number of turns in the coil; c_c is the speed of light within the coil; and ω_{\perp} is the angular rate as before.

A phase modulator is placed on the entrance to the coil for one light path and the exit for other. This introduces a time-dependent phase shift, such that light arriving at the detector simultaneously via the two paths is subject to different phase shifts. The phase-shift difference between the two paths, $\phi_p(t)$, is also time variant. By synchronizing the duty cycle of the detector with the phase modulator, samples can be taken at a particular value of ϕ_p . The intensity of the signal received at the detector is then

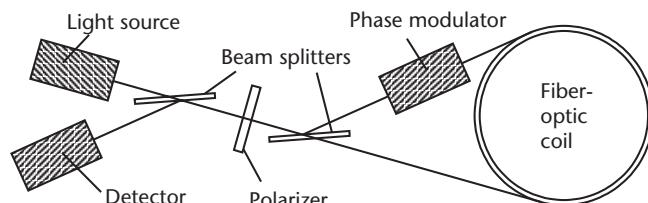


Figure 4.9 Interferometric fiber-optic gyro.

$$I_d = I_0[1 + \cos(\phi_c(\omega_{\perp}) + \phi_p(t))] \quad (4.5)$$

where I_0 is a constant. The scale factor of the intensity as a function of the rotation induced phase shift is

$$\frac{\partial I_d}{\partial \phi_c} = -I_0 \sin(\phi_c(\omega_{\perp}) + \phi_p(t)) \quad (4.6)$$

This is highly nonlinear and, without the phase modulator, gives zero scale factor for small angular rates. The sensitivity of the IFOG is optimized by selecting ϕ_p at the sampling time to maximize the scale factor. Best performance is obtained with closed-loop operation, whereby ϕ_p at the sampling time is constantly varied to keep the scale factor at its maximum value. The gyro sensitivity is also optimized by maximizing the coil diameter and number of turns. IFOGs are more reliable than both RLGs and spinning-mass gyros.

4.2.3 Vibratory Gyroscopes

A *vibratory gyroscope* comprises an element that is driven to undergo simple harmonic motion. The vibrating element may be a string, beam, pair of beams, tuning fork, ring, cylinder, or hemisphere. All operate on the same principle, which is to detect the Coriolis acceleration of the vibrating element when the gyro is rotated. This is easiest to illustrate with a vibrating string. Consider an element of the string, a , which oscillates about the center of the gyro body frame, b , at an angular frequency, ω_v . In the absence of both angular and linear motion of the gyro body, the dynamics of the string with respect to an inertial frame, i , with the same origin as the gyro body frame are described by the simple harmonic motion equations. Thus,

$$\begin{aligned} \mathbf{r}_{ia}^b \Big|_{\dot{\mathbf{r}}_{ib} = 0} &= \mathbf{r}_0^b \cos[\omega_v(t - t_0)] \\ \mathbf{v}_{ia}^b \Big|_{\dot{\mathbf{r}}_{ib} = 0} &= -\omega_v \mathbf{r}_0^b \sin[\omega_v(t - t_0)] \\ \mathbf{a}_{ia}^b \Big|_{\ddot{\mathbf{r}}_{ib} = 0} &= -\omega_v^2 \mathbf{r}_0^b \cos[\omega_v(t - t_0)] \end{aligned} \quad (4.7)$$

From (2.46) and (2.35), the motion of the string in the axes of the gyro body frame is given in the general case by

$$\ddot{\mathbf{r}}_{ia}^b = \ddot{\mathbf{C}}_i^b \mathbf{r}_{ia}^b + 2\dot{\mathbf{C}}_i^b \dot{\mathbf{r}}_{ia}^b + \mathbf{a}_{ia}^b \quad (4.8)$$

Substituting in (2.28), (2.31), and (2.35),

$$\ddot{\mathbf{r}}_{ia}^b = \boldsymbol{\Omega}_{ib}^b \boldsymbol{\Omega}_{ib}^b \mathbf{r}_{ia}^b - 2\boldsymbol{\Omega}_{ib}^b \mathbf{v}_{ia}^b + \mathbf{a}_{ia}^b \quad (4.9)$$

Finally, substituting in (4.7) gives

$$\ddot{\mathbf{r}}_{ia}^b = \{\boldsymbol{\Omega}_{ib}^b \boldsymbol{\Omega}_{ib}^b \cos[\omega_v(t - t_0)] + 2\omega_v \boldsymbol{\Omega}_{ib}^b \sin[\omega_v(t - t_0)] - \omega_v^2 \cos[\omega_v(t - t_0)]\} \mathbf{r}_0^b \quad (4.10)$$

where the first term is the centrifugal acceleration, the second term the Coriolis acceleration, and the final term the driving acceleration of the simple harmonic motion. Where the gyro undergoes linear motion with respect to an inertial frame, additional terms are introduced to (4.10). However, both these and the centrifugal acceleration can be neglected if the vibration rate is set sufficiently high.

The Coriolis acceleration instigates simple harmonic motion along the axis perpendicular to both the driven vibration and the projection of the angular rate vector, $\boldsymbol{\omega}_{ib}^b$, in the plane perpendicular to the driven vibration. The amplitude of this motion is proportional to the angular rate. Rotation about the vibration axis does not produce a Coriolis acceleration. In practice, the motion of the vibrating element is constrained along one of the axes perpendicular to the driven vibration, so only rotation about this input axis leads to oscillation in the output axis, mutually perpendicular to the input and driven axes. Figure 4.10 illustrates this.

How the output vibration is detected depends on the gyro architecture [1, 2]. For string and single-beam gyros, the vibration of the element itself must be detected. In double-beam and tuning-fork gyros, the two elements are driven in antiphase, so their Coriolis-induced vibration is also in antiphase. This induces an oscillating torsion in the stem, which may be detected directly or via a pair of pickoff tines. Ring, cylinder, and hemispherical resonators have four drive units placed at right angles and four detectors at intermediate points. When the gyro is not rotating, the detectors are at the nodes of the vibration mode, so no signal is detected. When angular rate is applied, the vibration mode is rotated about the input axis. Figure 4.11 illustrates this.

Most vibratory gyros are low-cost, low-performance devices, often using MEMS technology and with quartz giving better performance than silicon. The exception is the hemispherical resonator gyro (HRG), which can offer aviation grade performance. The HRG is light, compact, and operates in a vacuum, so has become popular for space applications [5].

4.3 Inertial Measurement Units

Figure 4.12 shows the main elements of a typical inertial measurement unit: accelerometers and gyroscopes, the IMU processor, a calibration-parameters store, a

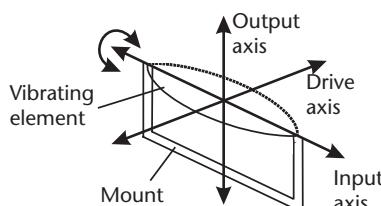


Figure 4.10 Axes of a vibrating gyro.

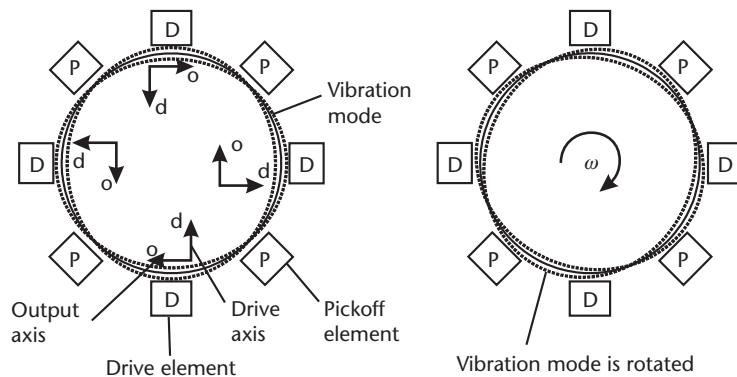


Figure 4.11 Vibration modes of ring, cylinder, and hemispherical vibratory gyros.

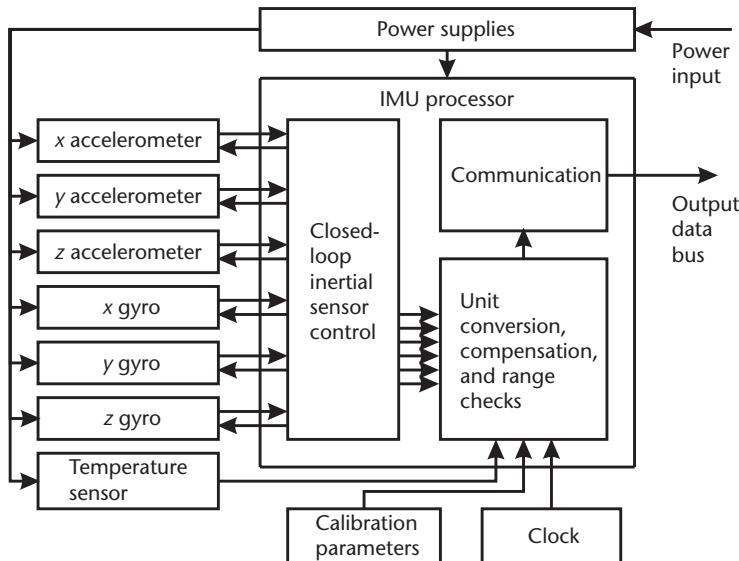


Figure 4.12 Schematic of an inertial measurement unit.

temperature sensor, and associated power supplies. Most IMUs have three accelerometers and three single-degree-of-freedom gyroscopes, mounted with orthogonal sensitive axes. However, some IMUs incorporate additional inertial sensors in a skewed configuration to protect against single sensor failure; this is discussed in Section 15.4.

The IMU processor performs unit conversion on the inertial sensor outputs, provides compensation for the known errors of the inertial sensors, and performs range checks to detect sensor failure. It may also incorporate closed-loop force feedback or rebalance control for the accelerometers and/or gyro. Unit conversion transforms the inertial sensor outputs from potential difference, current, or pulses into units of specific force and angular rate. Many IMUs integrate the specific force and angular rate over the sampling interval, τ , producing

$$\mathbf{v}_{ib}^b(t) = \int_{t-\tau}^t \mathbf{f}_{ib}^b(t') dt', \quad \boldsymbol{\alpha}_{ib}^b(t) = \int_{t-\tau}^t \boldsymbol{\omega}_{ib}^b(t') dt' \quad (4.11)$$

These are often referred to as “delta-v”s and “delta-θ”s. However, this can be misleading: the “delta-θ”s, $\boldsymbol{\alpha}_{ib}^b$, are attitude increments, but the “delta-v”s, \mathbf{v}_{ib}^b , are not velocity increments. The IMU outputs specific forces and angular rates, or their integrals, in the form of integers, which can be converted to SI units using scaling factors in the IMU’s documentation. Output rates typically vary between 100 and 1,000 Hz.

Inertial sensors exhibit constant errors that can be calibrated in the laboratory and stored in memory, enabling the IMU processor to correct the sensor outputs. Calibration parameters generally comprise accelerometer and gyro biases, scale factor and cross-coupling errors, and gyro g-dependent biases (see Section 4.4). These errors vary with temperature, so the calibration is performed at a range of temperatures and the IMU is equipped with a temperature sensor. However, the temperature within each individual sensor does not necessarily match the ambient temperature of the IMU, so some high-performance IMUs implement temperature control instead [1]. The cost of calibration may be minimized by applying the same set of calibration coefficients to a whole production batch of sensors. However, best performance is obtained by calibrating each sensor or IMU individually, noting that IMU-level calibration is needed to fully capture the cross-coupling errors. A Kalman filter may be used to obtain the calibration coefficients from the measurement data [11]. This process is known as laboratory calibration, to distinguish it from the in-run calibration discussed later.

A further source of accelerometer errors that the IMU processor compensates for is the *size effect*. To compute a navigation solution for a single point in space, the IMU’s angular rate and specific force measurements must also apply to a single reference point. However, in practice, the size of the inertial sensors demands that they are placed a few centimeters apart (generally less for MEMS sensors). Figure 4.13 illustrates this. For the gyros, this does not present a problem. However, rotation of an accelerometer about an axis perpendicular to its sensitive axis causes it to sense a centripetal force that is proportional to its distance from the axis of rotation. The centripetal force at the accelerometer is thus different to that at the reference point, resulting in an error in the measurement of the specific force at the reference point of

$$\Delta \mathbf{f}_{ib}^b = \begin{pmatrix} [(\omega_{ib,y}^b)^2 + (\omega_{ib,z}^b)^2] \Delta x_b \\ [(\omega_{ib,z}^b)^2 + (\omega_{ib,x}^b)^2] \Delta y_b \\ [(\omega_{ib,x}^b)^2 + (\omega_{ib,y}^b)^2] \Delta z_b \end{pmatrix} \quad (4.12)$$

where Δx_b , Δy_b , and Δz_b are the displacements of each accelerometer from the reference point along its sensitive axis. As the displacements are known from the

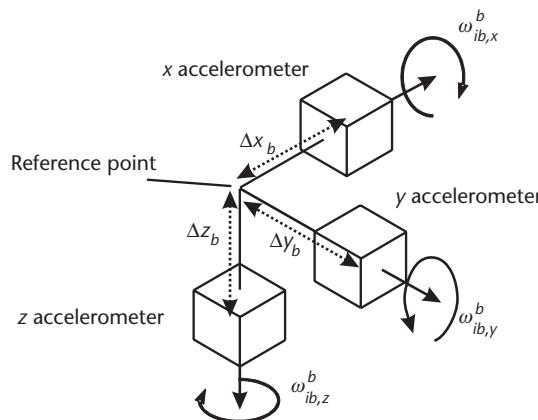


Figure 4.13 Accelerometer mounting relative to the IMU reference point. (From: [12]. © 2002 QinetiQ Ltd. Reprinted with permission.)

IMU design and the angular rates are measured by the gyros, this is easily compensated within the IMU processor.

4.4 Error Characteristics

All types of accelerometers and gyros exhibit biases, scale factor, and cross-coupling errors, and random noise to a certain extent. Higher order errors and angular rate-acceleration cross-sensitivity may also occur, depending on the sensor type. These are all discussed later.

Each systematic error source has four components: a fixed contribution, a temperature-dependent variation, a run-to-run variation, and an in-run variation. The fixed contribution is present each time the sensor is used and is corrected by the IMU processor using the laboratory calibration data. The temperature-dependent component can also be corrected by the IMU using laboratory calibration data. The run-to-run variation results in a contribution to the error source that is different each time the sensor is used but remains constant within any run. It cannot be corrected by the IMU processor, but it can be calibrated by the INS alignment and/or integration algorithms each time the IMU is used, as described in Section 5.5.3 and Chapters 12 and 13. Finally, the in-run variation contribution to the error source slowly changes during the course of a run. It cannot be corrected by the IMU or by an alignment process. In theory, it can be corrected through integration with other navigation sensors, but is difficult to observe in practice.

In discussing the error performance of different types and grades of inertial sensor here, the laboratory-calibrated contributions to the error sources, corrected within the IMU, are neglected, as it is the postcalibration performance of the inertial sensors that is relevant in determining inertial navigation performance and designing an integrated navigation system. Note that, as well as the run-to-run and in-run variations contributing to each error source, there are also residual fixed and temperature-dependent contributions left over from the calibration process.

4.4.1 Biases

The *bias* is a constant error exhibited by all accelerometers and gyros. It is independent of the underlying specific force and angular rate. It is sometimes called the g-independent bias to distinguish it from the g-dependent bias discussed in Section 4.4.4. In most cases, the bias is the dominant term in the overall error of an inertial instrument.

The accelerometer and gyro biases of an IMU, following sensor calibration and compensation, are denoted by the vectors $\mathbf{b}_a = (b_{a,x}, b_{a,y}, b_{a,z})$ and $\mathbf{b}_g = (b_{g,x}, b_{g,y}, b_{g,z})$, respectively. IMU errors are always expressed in body axes, so the superscript b may be omitted. Where the accelerometers and gyros form orthogonal triads, $b_{a,x}$ is the bias of the x -axis accelerometer (i.e., sensitive to specific force along the body frame x -axis), $b_{g,y}$ is the bias of the y -axis gyro, and so forth. For skewed-sensor configurations, the IMU biases may still be expressed as three-component vectors, but the components do not correspond to individual instruments.

It is sometimes convenient to split the biases into static, \mathbf{b}_{as} and \mathbf{b}_{gs} , and dynamic, \mathbf{b}_{ad} and \mathbf{b}_{gd} , components, where

$$\mathbf{b}_a = \mathbf{b}_{as} + \mathbf{b}_{ad} \quad \mathbf{b}_g = \mathbf{b}_{gs} + \mathbf{b}_{gd} \quad (4.13)$$

The static component, also known as the fixed bias, turn-on bias or bias repeatability, comprises the run-to-run variation of each instrument bias plus the residual fixed bias remaining after sensor calibration. It is constant throughout an IMU operating period, but varies from run to run. The dynamic component, also known as the in-run bias variation or bias instability, varies over periods of order a minute and also incorporates the residual temperature-dependent bias remaining after sensor calibration. The dynamic bias is typically about 10 percent of the static bias.

Accelerometer and gyro biases are not usually quoted in SI units. Accelerometer biases are quoted in terms of the acceleration due to gravity, abbreviated to g, where $1\text{ g} = 9.80665\text{ m s}^{-2}$ [13], noting that the actual acceleration due to gravity varies with location (see Section 2.3.5). Units of milli-g (mg) or micro-g (μg) are used. For gyro biases, degrees per hour (${}^\circ\text{ hr}^{-1}$ or deg/hr) are used where $1{}^\circ\text{ hr}^{-1} = 4.848 \times 10^{-6}\text{ rad s}^{-1}$, except for very poor-quality gyros, where degrees per second are used. Table 4.1 gives typical accelerometer and gyro biases for different grades of IMU [1, 5].

Table 4.1 Typical Accelerometer and Gyro Biases for Different Grades of IMU

IMU Grade	Accelerometer Bias mg	Accelerometer Bias m s^{-2}	Gyro Bias ${}^\circ\text{ hr}^{-1}$	Gyro Bias rad s^{-1}
Marine	0.01	10^{-4}	0.001	5×10^{-9}
Aviation	0.03–0.1	$3 \times 10^{-4} - 10^{-3}$	0.01	5×10^{-8}
Intermediate	0.1–1	$10^{-3} - 10^{-2}$	0.1	5×10^{-7}
Tactical	1–10	$10^{-2} - 10^{-1}$	1–100	$5 \times 10^{-6} - 5 \times 10^{-4}$
Automotive	>10	> 10^{-1}	>100	> 5×10^{-4}

Pendulous accelerometers span most of the performance range, while VBAs exhibit biases of 0.1 mg upward. MEMS accelerometers using both technologies exhibit the largest biases. Ring laser gyros exhibit biases as low as $0.001^\circ \text{ hr}^{-1}$. However, low-cost RLGs can exhibit biases up to 10° hr^{-1} . IFOGs typically exhibit biases between 0.01 and $100^\circ \text{ hr}^{-1}$. Vibratory-gyro biases range from 1° hr^{-1} to 1° s^{-1} , while spinning-mass gyros span the whole performance range.

4.4.2 Scale Factor and Cross-Coupling Errors

The *scale factor error* is the departure of the input-output gradient of the instrument from unity following unit conversion by the IMU. Figure 4.14 illustrates this. The accelerometer output error due to the scale factor error is proportional to the true specific force along the sensitive axis, while the gyro output error due to the scale factor error is proportional to the true angular rate about the sensitive axis. The accelerometer and gyro scale factor errors of an IMU are denoted by the vectors $s_a = (s_{a,x}, s_{a,y}, s_{a,z})$ and $s_g = (s_{g,x}, s_{g,y}, s_{g,z})$, respectively.

Cross-coupling errors in all types of IMU arise from the misalignment of the sensitive axes of the inertial sensors with respect to the orthogonal axes of the body frame due to manufacturing limitations as illustrated in Figure 4.15. Hence, some authors describe these as misalignment errors. These make each accelerometer

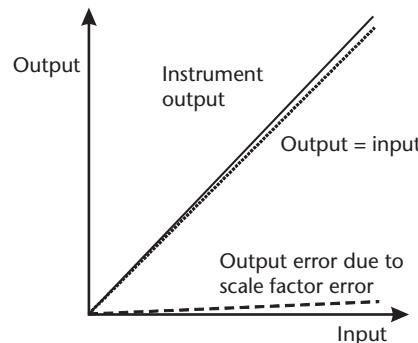


Figure 4.14 Scale factor error. (From: [12]. © 2002 QinetiQ Ltd. Reprinted with permission.)

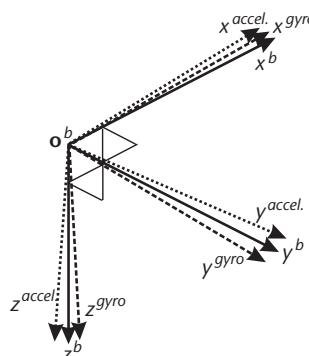


Figure 4.15 Misalignment of accelerometer and gyro sensitive axes with respect to the body frame.

sensitive to the specific force along the axes orthogonal to its sensitive axis and each gyro sensitive to the angular rate about the axes orthogonal to its sensitive axis. The axes misalignment also produces additional scale factor errors, but these are typically two to four orders of magnitude smaller than the cross-coupling errors. In vibratory sensors, cross-coupling errors can also arise due to the cross-talk between the individual sensors. The notation $m_{a,\alpha\beta}$ is used to denote the cross-coupling coefficient of β -axis specific force sensed by the α -axis accelerometer, while $m_{g,\alpha\beta}$ denotes the coefficient of β -axis angular rate sensed by the α -axis gyro.

The scale factor and cross-coupling errors for a nominally orthogonal accelerometer and gyro triad may be expressed as the following matrices:

$$\mathbf{M}_a = \begin{pmatrix} s_{a,x} & m_{a,xy} & m_{a,xz} \\ m_{a,yx} & s_{a,y} & m_{a,yz} \\ m_{a,zx} & m_{a,zy} & s_{a,z} \end{pmatrix} \quad \mathbf{M}_g = \begin{pmatrix} s_{g,x} & m_{g,xy} & m_{g,xz} \\ m_{g,yx} & s_{g,y} & m_{g,yz} \\ m_{g,zx} & m_{g,zy} & s_{g,z} \end{pmatrix} \quad (4.14)$$

The total specific force and angular rate measurement errors due to the scale factor and cross-coupling errors are then $\mathbf{M}_a \mathbf{f}_{ib}^b$ and $\mathbf{M}_g \boldsymbol{\omega}_{ib}^b$, respectively. Scale factor and cross-coupling errors are **unitless** and typically expressed in parts per million (ppm) or as a percentage. Some manufacturers quote the axis misalignments instead of the cross-coupling errors, noting that the latter is the sine of the former.

Where the cross-coupling errors arise only from axis misalignment, 3 of the 12 components may be eliminated by defining the body-frame axes in terms of the sensitive axes of the inertial sensors. One convention is to define the body-frame z -axis as the sensitive axis of the z gyro and the body-frame y -axis such that the sensitive axis of the y gyro lies in the yz plane. This eliminates $m_{g,zx}$, $m_{g,zy}$, and $m_{g,yx}$.

For most inertial sensors, the scale factor and cross-coupling errors are between 10^{-4} and 10^{-3} (100–1,000 ppm). The main exceptions are some MEMS gyros where these errors can be as high as 10^{-2} (1 percent) and ring laser gyros, which exhibit low scale factor errors, typically between 10^{-5} and 10^{-4} (10–100 ppm). The lowest cost sensors can exhibit significant scale factor asymmetry, whereby the scale factor errors are different for positive and negative readings.

4.4.3 Random Noise

All inertial sensors exhibit random noise from a number of sources. Electrical noise limits the resolution of inertial sensors, particularly MEMS sensors, where the signal is very weak. Pendulous accelerometers and spinning-mass gyros exhibit noise due to mechanical instabilities, while the residual lock-in effects of an RLG, after dithering is applied, manifest as noise [1]. VBAs and vibratory gyros can exhibit high-frequency resonances. In addition, vibration from spinning-mass gyros and RLG dither motors can induce accelerometer noise [14]. The random noise on each IMU sample is denoted by the vectors $\mathbf{w}_a = (w_{a,x}, w_{a,y}, w_{a,z})$ and $\mathbf{w}_g = (w_{g,x}, w_{g,y}, w_{g,z})$ for the accelerometers and gyros, respectively.

The spectrum of accelerometer and gyro noise for frequencies below 1 Hz is approximately white, so the standard deviation of the average specific force and angular rate noise varies in inverse proportion to the square root of the averaging time. Inertial sensor noise is thus usually quoted in terms of the root PSD, n . The customary units are $\mu\text{g}/\sqrt{\text{Hz}}$ for accelerometer random noise, where $1 \mu\text{g}/\sqrt{\text{Hz}} = 9.80665 \times 10^{-5} \text{ m s}^{-1.5}$, and $^\circ/\sqrt{\text{hr}}$ or $^\circ/\text{hr}/\sqrt{\text{Hz}}$ for gyro random noise, where $1^\circ/\sqrt{\text{hr}} = 2.909 \times 10^{-4} \text{ rad s}^{-0.5}$ and $1^\circ/\text{hr}/\sqrt{\text{Hz}} = 4.848 \times 10^{-6} \text{ rad s}^{-0.5}$. The standard deviation of the random noise samples are obtained by multiplying the corresponding root PSDs by the root of the sampling rate or dividing them by the root of the sampling interval. White random noise cannot be calibrated and compensated, as there is no correlation between past and future values.

MEMS sensors can also exhibit significant high-frequency noise [15]. Within the IMU body frame, this noise averages out over the order of a second. However, if the IMU is rotating, the noise will not average out to the same extent within the frame used to compute the inertial navigation solution. Consequently, caution should be exercised in selecting these sensors for highly dynamic applications. Problems can also arise in a vibration environment. Where the frequency of the external vibration is close to one of the inertial sensor's resonant frequencies, time-correlated errors will ensue. Otherwise, the interaction between the vibration and the high-frequency noise will cause an increase in the effective white noise exhibited by the sensor. High-frequency noise can potentially be reduced using wavelet filtering techniques [16] or an artificial neural network (ANN) [17]. However, these techniques reduce the effective sensor bandwidth.

The accelerometer and gyro random noise are sometimes described as random walks, which can be a cause of confusion. Random noise on the specific force measurements is integrated to produce a random-walk error on the inertial velocity solution. Similarly, random noise on the angular rate measurements is integrated to produce an attitude random-walk error. The standard deviation of a random-walk process is proportional to the square root of the integration time. The same random-walk errors are obtained by summing the random noise on integrated specific force and attitude increment IMU outputs.

A further source of noise is the quantization of the IMU data-bus outputs. Word lengths of 16 bits are typically used for the integrated specific force and attitude increment outputs of a tactical-grade IMU, \boldsymbol{v}_{ib}^b and $\boldsymbol{\alpha}_{ib}^b$, giving quantization levels of the order of 10^{-4} m s^{-1} and $2 \times 10^{-6} \text{ rad}$, respectively. The IMU's internal processor generally operates to a higher precision, so the residuals are carried over to the next iteration. Consequently, the standard deviation of the quantization noise averaged over successive IMU outputs varies in inverse proportion to the number of samples, rather than the square root, until the IMU internal quantization limit is reached.

The accelerometer random-noise root PSD varies from about $20 \mu\text{g}/\sqrt{\text{Hz}}$ for aviation-grade IMUs, through about $100 \mu\text{g}/\sqrt{\text{Hz}}$ for tactical-grade IMUs using pendulous accelerometers or quartz VBAs to the order of $1,000 \mu\text{g}/\sqrt{\text{Hz}}$ for silicon VBAs. Spinning-mass gyros exhibit the smallest random noise with a root PSD of $0.002^\circ/\sqrt{\text{hr}}$ typical of aviation-grade gyros, while RLGs exhibit random noise about

an order of magnitude higher. Tactical-grade IMUs using IFOGs or quartz vibratory gyros typically exhibit a gyro random noise root PSD in the $0.03\text{--}0.1^\circ/\sqrt{\text{hr}}$ range. The root PSD for the random noise of silicon vibratory gyros is typically $1^\circ/\sqrt{\text{hr}}$ or more.

4.4.4 Further Error Sources

Accelerometers and gyros exhibit further error characteristics depending on the sensor design.

Spinning mass and vibratory gyros exhibit a sensitivity to specific force, known as the *g-dependent bias*, due to mass unbalance. Some designs of IFOG also exhibit a g-dependent bias. It is typically of the order of $1^\circ/\text{hr/g}$ (4.944×10^{-5} rad m $^{-1}$ s) but can be as high as $100^\circ/\text{hr/g}$ for uncompensated automotive-grade MEMS gyros. Gyros can be sensitive to accelerations along all three axes, so the g-dependent bias for a gyro triad comprises the 3×3 matrix, \mathbf{G}_g .

Inertial sensors can exhibit scale factor nonlinearity, as distinct from scale factor errors, whereby the scale factor varies with the specific force or angular rate. The nonlinearity is expressed as the variation of the scale factor over the operating range of the sensor. Note that this figure does not describe the shape of the scale factor variation, which can range from linear to irregular and need not be symmetric about the zero point. However, the larger departures from scale factor linearity occur at high angular rates and specific forces. The scale factor nonlinearity is generally of a similar order to the scale factor error, ranging from 10^{-5} for some RLGs, through 10^{-4} to 10^{-3} for most inertial sensors, to 10^{-2} for some MEMS gyros.

Spinning-mass gyros and pendulous accelerometers also exhibit a number of higher order errors [1]. The gyros can exhibit anisoelastic or g^2 -dependent biases, which are proportional to the product of accelerations along pairs of orthogonal axes, and anisoinertia errors, which are proportional to the product of angular rate about pairs of orthogonal axes. Pendulous accelerometers can exhibit vibropendulous errors, which are proportional to the product of accelerations along the sensitive and pendulum axes. They are also sensitive to angular acceleration about the hinge axis with the hinge-proof mass separation forming the constant of proportionality [18].

4.4.5 Error Models

The following equations show how the main error sources contribute to the accelerometer and gyro outputs:

$$\tilde{\mathbf{f}}_{ib}^b = \mathbf{b}_a + (\mathbf{I}_3 + \mathbf{M}_a) \mathbf{f}_{ib}^b + \mathbf{w}_a \quad (4.15)$$

$$\tilde{\boldsymbol{\omega}}_{ib}^b = \mathbf{b}_g + (\mathbf{I}_3 + \mathbf{M}_g) \boldsymbol{\omega}_{ib}^b + \mathbf{G}_g \mathbf{f}_{ib}^b + \mathbf{w}_g \quad (4.16)$$

where $\tilde{\mathbf{f}}_{ib}^b$ and $\tilde{\boldsymbol{\omega}}_{ib}^b$ are the IMU-output specific force and angular rate vectors, \mathbf{f}_{ib}^b and $\boldsymbol{\omega}_{ib}^b$ are the true counterparts, and \mathbf{I}_3 is the identity matrix. The total accelerometer and gyro errors are

$$\begin{aligned}\delta \mathbf{f}_{ib}^b &= \tilde{\mathbf{f}}_{ib}^b - \mathbf{f}_{ib}^b \\ \delta \boldsymbol{\omega}_{ib}^b &= \tilde{\boldsymbol{\omega}}_{ib}^b - \boldsymbol{\omega}_{ib}^b\end{aligned}\quad (4.17)$$

Where estimates of the biases, scale factor and cross-coupling errors, and gyro g-dependent errors are available, corrections may be applied:

$$\begin{aligned}\hat{\mathbf{f}}_{ib}^b &= (\mathbf{I}_3 + \hat{\mathbf{M}}_a)^{-1} \tilde{\mathbf{f}}_{ib}^b - \hat{\mathbf{b}}_a \\ \hat{\boldsymbol{\omega}}_{ib}^b &= (\mathbf{I}_3 + \hat{\mathbf{M}}_g)^{-1} \tilde{\boldsymbol{\omega}}_{ib}^b - \hat{\mathbf{b}}_g - \hat{\mathbf{G}}_g \hat{\mathbf{f}}_{ib}^b\end{aligned}\quad (4.18)$$

where the carat, $\hat{\cdot}$, is used to denote an estimate and, applying a power-series expansion,

$$(\mathbf{I}_3 + \hat{\mathbf{M}}_{a/g})^{-1} = \mathbf{I}_3 + \sum_r \binom{-1}{r} \hat{\mathbf{M}}_{a/g}^r \approx \mathbf{I}_3 - \hat{\mathbf{M}}_{a/g} + \hat{\mathbf{M}}_{a/g}^2 \quad (4.19)$$

A similar formulation is used for applying the laboratory calibration within the IMU processor, noting that, in that case, the error measurements are functions of temperature.

References

- [1] Titterton, D. H., and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., Stevenage, U.K.: IEE, 2004.
- [2] Lawrence, A., *Modern Inertial Technology*, 2nd ed., New York: Springer-Verlag, 2001.
- [3] Norgia, M., and S. Donati, "Hybrid Opto-Mechanical Gyroscope with Injection-Interferometer Readout," *Electronics Letters*, Vol. 37, No. 12, 2001, pp. 756–758.
- [4] El-Sheimy, N., and X. Niu, "The Promise of MEMS to the Navigation Community," *Inside GNSS*, March–April 2007, pp. 46–56.
- [5] Barbour, N. M., "Inertial Navigation Sensors," *Advances in Navigation Sensors and Integration Technology*, NATO RTO Lecture Series-232, London, U.K., October 2003, paper 2.
- [6] Jekeli, C., "Cold Atom Interferometer as Inertial Measurement Unit for Precision Navigation," *Proc. ION 60th AM*, Dayton, OH, June 2004, pp. 604–613.
- [7] Sorg, H. W., "From Serson to Draper—Two Centuries of Gyroscopic Development," *Navigation: JION*, Vol. 23, No. 4, 1976, pp. 313–323.
- [8] Macek, W. M., and D. T. M. Davis, "Rotation Rate Sensing with Traveling-Wave Ring Lasers," *Applied Physics Letters*, Vol. 2, No. 5, 1963, pp. 67–68.
- [9] Vali, V., and R. W. Shorthill, "Fiber Ring Interferometer," *Applied Optics*, Vol. 15, No. 15, 1976, pp. 1099–1100.
- [10] Matthews, A., "Utilization of Fiber Optic Gyros in Inertial Measurement Units," *Navigation: JION*, Vol. 27, No. 1, 1990, pp. 17–38.
- [11] Fountain, J. R., "Silicon IMU for Missile and Munitions Applications," *Advances in Navigation Sensors and Integration Technology*, NATO RTO Lecture Series-232, London, U.K., October 2003, paper 10.
- [12] Groves, P. D., "Principles of Integrated Navigation," Course Notes, QinetiQ Ltd., 2002.

- [13] Tennent, R. M., *Science Data Book*, Edinburgh, U.K.: Oliver & Boyd, 1971.
- [14] Woolven, S., and D. B. Reid, "IMU Noise Evaluation for Attitude Determination and Stabilization in Ground and Airborne Applications," *Proc. IEEE PLANS*, Las Vegas, NV, April 1994, pp. 817–822.
- [15] Fountain, J. R., "Characteristics and Overview of a Silicon Vibrating Structure Gyroscope," *Advances in Navigation Sensors and Integration Technology*, NATO RTO Lecture Series-232, London, U.K., October 2003, paper 8.
- [16] Shalard, J., A. M. Bruton, and K. P. Schwarz, "Detection and Filtering of Short Term ($1/f^{\gamma}$) Noise in Inertial Sensors," *Navigation: JION*, Vol. 46, No. 2, 1999, pp. 97–107.
- [17] El-Rabbany, A., and M. El-Diasty, "An Efficient Neural Network Model for De-Noising of MEMS-Based Inertial Data," *Journal of Navigation*, Vol. 57, No. 3, 2004, pp. 407–415.
- [18] Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, New York: John Wiley & Sons, 2001.

Inertial Navigation

An inertial navigation system (INS), sometimes known as an inertial navigation unit (INU), is a dead-reckoning navigation system, comprising an inertial measurement unit and a navigation processor, as shown in Figure 1.3. The IMU, described in the last chapter, incorporates a set of accelerometers and gyros and, assuming a strapdown configuration, produces measurements of the specific force, \mathbf{f}_{ib}^b , and angular rate, $\boldsymbol{\omega}_{ib}^b$, of the body frame with respect to inertial space in body-frame axes. Alternatively, it may output integrated specific force, \mathbf{v}_{ib}^b , and attitude increments, $\boldsymbol{\alpha}_{ib}^b$.

This chapter focuses on the navigation processor. This may be packaged with the IMU and the system sold as a complete INS. Alternatively, the navigation equations may be implemented on an integrated navigation processor or on the application's central processor. Marine, aviation, and intermediate grade inertial sensors tend to be sold as part of an INS, while tactical grade inertial sensors are usually sold as an IMU. In either case, the function is the same, so the term inertial navigation system is applied here to all architectures where a three-dimensional navigation solution is obtained from inertial sensor measurements. Other uses of inertial sensors in navigation and dead reckoning techniques using other types of sensor are described in Chapter 10.

Figure 5.1 shows a schematic of an inertial navigation processor. This integrates the IMU outputs to produce a position, velocity, and attitude solution. The navigation equations comprise four steps: attitude update, transformation of the specific-force resolving axes, velocity update, and position update. In addition, a gravity or gravitation model is needed to transform specific force into acceleration (see Section 2.3.5). In basic terms, the attitude is updated by integrating the angular-rate measurements; the velocity is updated by integrating acceleration; and the position is updated by integrating velocity. In an integrated navigation system, there may also be correction of the IMU outputs and the inertial navigation solution using estimates from the integration algorithm (see Section 12.1.1).

The form of the inertial navigation equations depends on which coordinate frame the navigation solution is expressed in. Section 5.1 describes the navigation equations for the ECI frame implementation, while Section 5.2 describes how they are modified for implementation in the rotating ECEF frame. Section 5.3 presents the Earth-referenced local navigation frame implementation with curvilinear position and discusses the wander azimuth frame variant. Note that the navigation solution does not have to be computed in the same frame that is used for user output. Transformation of the navigation solution between coordinate frames was

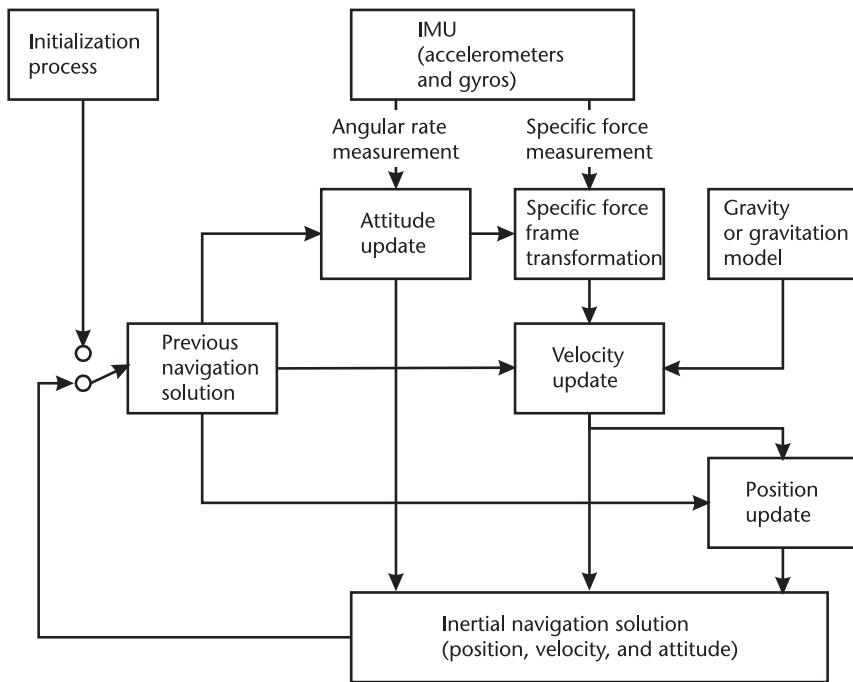


Figure 5.1 Schematic of an inertial navigation processor.

described in Section 2.4, while Section 2.3.2 described the transformation between Cartesian and curvilinear position. The navigation equations presented in Sections 5.1 to 5.3 are the simplest practical form, making use of a number of first-order approximations and assuming that the navigation equations are iterated at the IMU output rate. Section 5.4 describes how the precision of the inertial navigation equations may be improved and discusses which forms are appropriate for different applications. All the navigation equations presented here use the coordinate transformation matrix representation of attitude, as this is the clearest. The quaternion form of the attitude update is described in [1, 2].

Computation of an inertial navigation solution is an iterative process, making use of the solution from the previous iteration. Therefore, the navigation solution must be initialized before the INS can function. Section 5.5 describes the different methods of initializing the position, velocity, and attitude, including self-alignment and fine alignment processes.

Section 5.6 describes the error behavior of an INS. Errors can arise from the IMU, the initialization process, and the navigation equations. These then propagate through the navigation equations to give position, velocity, and attitude errors that vary with time. The short-term and long-term cases are examined. Finally, Section 5.7 discusses the platform INS, and Section 5.8 discusses horizontal-plane inertial navigation.

5.1 Inertial-Frame Navigation Equations

Figure 5.2 shows how the angular-rate and specific-force measurements made over the time interval t to $t + \tau_i$ are used to update the attitude, velocity, and position,

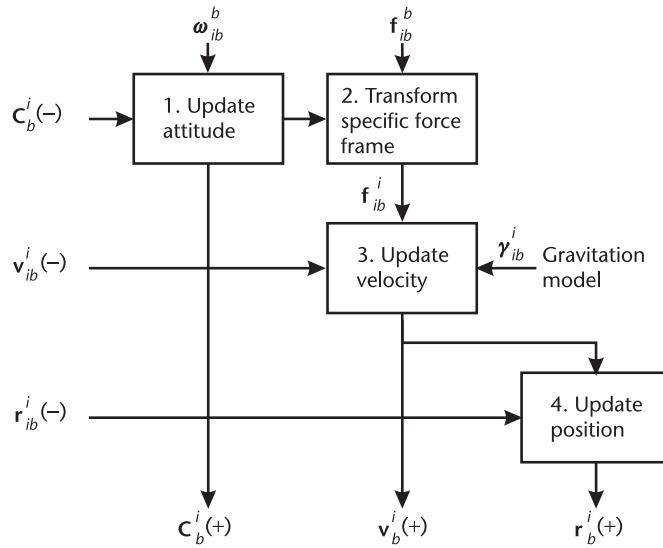


Figure 5.2 Block diagram of ECI-frame navigation equations.

expressed with respect to resolved and in the axes of the ECI coordinate frame. Each of the four steps is described in turn. The suffixes (–) and (+) are, respectively, used to denote values at the beginning of the navigation equations processing cycle, at time t , and at the end of the processing cycle, at time $t + \tau_i$. The inertial-frame navigation equations are the simplest of those presented here. However, a frame transformation must be applied to obtain an Earth-referenced solution for user output.

5.1.1 Attitude Update

The attitude update step of the inertial navigation equations uses the angular-rate measurement from the IMU, ω_{ib}^b , to update the attitude solution, expressed as the body-to-inertial-frame coordinate transformation matrix, C_b^i .

From (2.28), the time derivative of the coordinate transformation matrix is

$$\dot{C}_b^i = C_b^i \Omega_{ib}^b \quad (5.1)$$

recalling from Section 2.2.5 that $\Omega_{ib}^b = [\omega_{ib}^b \wedge]$, the skew-symmetric matrix of the angular rate. Integrating this gives

$$C_b^i(t + \tau_i) = C_b^i(t) \exp\left(\int_t^{t + \tau_i} \Omega_{ib}^b dt\right) \quad (5.2)$$

Applying (4.11), this may be expressed in terms of the attitude increment, α_{ib}^b :

$$\mathbf{C}_b^i(t + \tau_i) = \mathbf{C}_b^i(t) \exp([\alpha_{ib}^b \wedge]) \quad (5.3)$$

The exponent of a matrix is not the same as the matrix of the exponents of its components. Expressing (5.3) as a power series,

$$\mathbf{C}_b^i(t + \tau_i) = \mathbf{C}_b^i(t) \sum_{r=0}^{\infty} \frac{[\alpha_{ib}^b \wedge]^r}{r!} \quad (5.4)$$

The simplest form of the attitude update is obtained by truncating the power-series expansion to first order:

$$\mathbf{C}_b^i(+) \approx \mathbf{C}_b^i(-) (\mathbf{I}_3 + [\alpha_{ib}^b \wedge]) \quad (5.5)$$

Where the angular rate is assumed to be constant over the attitude integration interval, $\alpha_{ib}^b \approx \omega_{ib}^b \tau_i$. This always applies where the attitude integration is performed at the IMU output rate. In this case, (5.5) becomes

$$\mathbf{C}_b^i(+) \approx \mathbf{C}_b^i(-) (\mathbf{I}_3 + \boldsymbol{\Omega}_{ib}^b \tau) \quad (5.6)$$

where

$$\mathbf{I}_3 + \boldsymbol{\Omega}_{ib}^b \tau_i = \begin{pmatrix} 1 & -\omega_{ib,z}^b \tau_i & \omega_{ib,y}^b \tau_i \\ \omega_{ib,z}^b \tau_i & 1 & -\omega_{ib,x}^b \tau_i \\ -\omega_{ib,y}^b \tau_i & \omega_{ib,x}^b \tau_i & 1 \end{pmatrix} \quad (5.7)$$

The first-order approximation of (5.4) is a form of the small angle approximation, $\sin \theta \approx \theta$, $\cos \theta \approx 1$. Clearly, the truncation of the power series will introduce errors in the attitude integration, which will be larger at lower iteration rates (large τ_i) and higher angular rates. Precision may be improved at the expense of increased complexity and processor load by including higher order terms in the power series, (5.4), or applying the exact solution as described in Section 5.4.2. Alternatively, the integration may be broken down into smaller steps as discussed Section 5.4.1.³

5.1.2 Specific-Force Frame Transformation

The IMU measures specific force along the body-frame resolving axes. However, for use in the velocity integration step of the navigation equations, it must be resolved about the same axes as the velocity—in this case, the ECI frame. The resolving axes are transformed simply by applying a coordinate transformation matrix:

$$\mathbf{f}_{ib}^i(t) = \mathbf{C}_b^i(t) \mathbf{f}_{ib}^b(t) \quad (5.8)$$

As the specific-force measurement is an average over time t to $t + \tau_i$, the coordinate transformation matrix should be similarly averaged. A simple implementation is¹

$$\mathbf{f}_{ib}^i \approx \frac{1}{2} (\mathbf{C}_b^i(-) + \mathbf{C}_b^i(+)) \mathbf{f}_{ib}^b \quad (5.9)$$

However, it should be noted that the mean of two coordinate transformation matrices does not precisely produce the mean of the two attitudes. A more accurate form is presented in Section 5.4.3. The less the attitude varies over the time interval, the smaller the errors introduced by this approximation.²

Where the IMU outputs integrated specific force, this is transformed in the same way:

$$\mathbf{v}_{ib}^i = \bar{\mathbf{C}}_b^i \mathbf{v}_{ib}^b, \quad \mathbf{v}_{ib}^i \approx \frac{1}{2} (\mathbf{C}_b^i(-) + \mathbf{C}_b^i(+)) \mathbf{v}_{ib}^b \quad (5.10)$$

where $\bar{\mathbf{C}}_b^i$ is the average value of the coordinate transformation matrix over the interval from t to $t + \tau_i$.

5.1.3 Velocity Update

As given by (2.76), inertially referenced acceleration is obtained simply by adding the gravitational acceleration to the specific force:

$$\mathbf{a}_{ib}^i = \mathbf{f}_{ib}^i + \boldsymbol{\gamma}_{ib}^i(\mathbf{r}_{ib}^i) \quad (5.11)$$

where (2.91) models the gravitational acceleration, $\boldsymbol{\gamma}_{ib}^i$, as a function of Cartesian position in the ECI frame. Strictly, the position should be averaged over the interval t to $t + \tau_i$. However, this would require recursive navigation equations, and the gravitational field varies slowly with position, so it is generally sufficient to use³ $\mathbf{r}_{ib}^i(-)$.

Where the reference frame and resolving axes are the same, the time derivative of velocity is simply acceleration, as shown by (2.45). Thus,

$$\dot{\mathbf{v}}_{ib}^i = \mathbf{a}_{ib}^i \quad (5.12)$$

Where variations in the acceleration over the velocity update interval are not known, as is the case where the velocity integration is iterated at the IMU output rate, the velocity update equation, obtained by integrating (5.12), is simply³

$$\mathbf{v}_{ib}^i(+) = \mathbf{v}_{ib}^i(-) + \mathbf{a}_{ib}^i \tau_i \quad (5.13)$$

From (4.11), (5.11), and (5.12), the velocity update in terms of integrated specific force is

$$\mathbf{v}_{ib}^i(+) = \mathbf{v}_{ib}^i(-) + \boldsymbol{\omega}_{ib}^b + \boldsymbol{\gamma}_{ib}^b \tau_i \quad (5.14)$$

5.1.4 Position Update

In the inertial-frame implementation of the navigation equations, the time derivative of the Cartesian position is simply velocity as the reference frame and resolving axes are the same (see (2.36)). Thus,

$$\dot{\mathbf{r}}_{ib}^i = \mathbf{v}_{ib}^i \quad (5.15)$$

In the velocity update step where the variation in acceleration is unknown, \mathbf{v}_{ib}^i is modeled as a linear function of time over the interval t to $t + \tau_i$. Integrating (5.15) thus leads to the position being modeled as a quadratic function of time. The velocity is known at the start and finish of the update interval, so the position is updated using

$$\begin{aligned} \mathbf{r}_{ib}^i(+) &= \mathbf{r}_{ib}^i(-) + (\mathbf{v}_{ib}^i(-) + \mathbf{v}_{ib}^i(+)) \frac{\tau_i}{2} \\ &= \mathbf{r}_{ib}^i(-) + \mathbf{v}_{ib}^i(-) \tau_i + \mathbf{a}_{ib}^i \frac{\tau_i^2}{2} \\ &= \mathbf{r}_{ib}^i(-) + \mathbf{v}_{ib}^i(+)\tau_i - \mathbf{a}_{ib}^i \frac{\tau_i^2}{2} \end{aligned} \quad (5.16)$$

where the three implementations are equally valid.³

5.2 Earth-Frame Navigation Equations

The ECEF frame is commonly used as the reference frame and resolving axes for computation of satellite navigation solutions (see Section 7.5.1), so, in an integrated system, there are benefits in using the same frame for computation of the inertial navigation solution. For some applications, such as airborne photogrammetry, the final navigation solution is more conveniently expressed in the ECEF frame [3]. A disadvantage of the ECEF-frame implementation compared to the inertial-frame implementation is that the rotation of the reference frame used for navigation solution computation with respect to the inertial reference, used for the inertial sensor measurements, introduces additional complexity. Figure 5.3 is a block diagram showing how the angular-rate and specific-force measurements are used to update the Earth-referenced attitude, velocity, and position. Each of the four steps is described in turn.

5.2.1 Attitude Update

The attitude update step of the ECEF-frame navigation equations uses the angular-rate measurement, $\boldsymbol{\omega}_{ib}^b$, to update the attitude solution, expressed as the body-to-

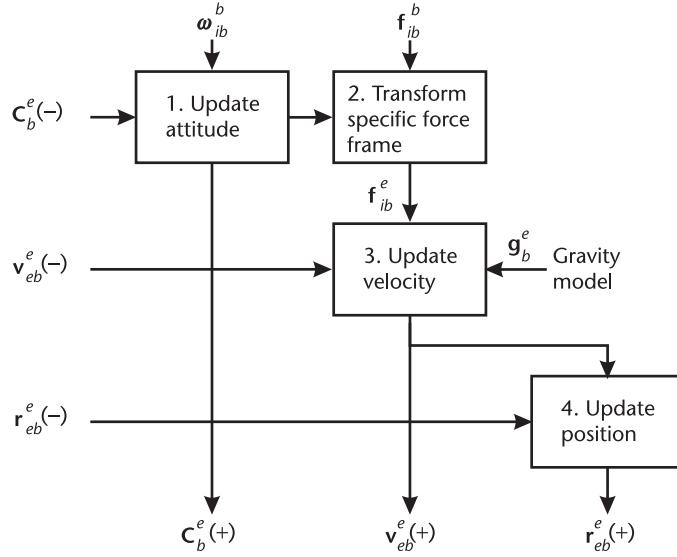


Figure 5.3 Block diagram of ECEF-frame navigation equations.

Earth-frame coordinate transformation matrix, \mathbf{C}_b^e . From (2.28), (2.25), and (2.27), the time derivative is

$$\begin{aligned}\dot{\mathbf{C}}_b^e &= \mathbf{C}_b^e \boldsymbol{\Omega}_{eb}^b \\ &= \mathbf{C}_b^e \boldsymbol{\Omega}_{ib}^b - \boldsymbol{\Omega}_{ie}^e \mathbf{C}_b^e\end{aligned}\quad (5.17)$$

where $\boldsymbol{\Omega}_{ib}^b$ is the skew-symmetric matrix of the IMU's angular-rate measurement, and $\boldsymbol{\Omega}_{ie}^e$ is the skew-symmetric matrix of the Earth-rotation vector. Thus, the rotation of the Earth must be accounted for in updating the attitude. From (2.74),

$$\boldsymbol{\Omega}_{ie}^e = \begin{pmatrix} 0 & -\omega_{ie} & 0 \\ \omega_{ie} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad (5.18)$$

Integrating (5.17) gives

$$\mathbf{C}_b^e(t + \tau_i) = \mathbf{C}_b^e(t) \exp([\boldsymbol{\alpha}_{ib}^b \wedge]) - [\exp(\boldsymbol{\Omega}_{ie}^e \tau_i) - \mathbf{I}_3] \mathbf{C}_b^e(t) \quad (5.19)$$

As in the ECI frame implementation, the exponents must be computed as power-series expansions. Applying the small angle approximation by truncating the expansions at first order and assuming the IMU angular-rate measurement is constant over the integration interval (i.e., $\boldsymbol{\alpha}_{ib}^b \approx \boldsymbol{\omega}_{ib}^b \tau_i$) gives

$$\mathbf{C}_b^e(+) \approx \mathbf{C}_b^e(-) (\mathbf{I}_3 + \boldsymbol{\Omega}_{ib}^b \tau_i) - \boldsymbol{\Omega}_{ie}^e \mathbf{C}_b^e(-) \tau_i \quad (5.20)$$

As the Earth rotation rate is very slow compared to the angular rates measured by the IMU, this small angle approximation is always valid for the Earth rate term of the attitude update equation.

5.2.2 Specific-Force Frame Transformation

The specific-force frame transformation takes the same form as in the inertial-frame implementation:

$$\begin{aligned}\mathbf{f}_{ib}^e(t) &= \mathbf{C}_b^e(t) \mathbf{f}_{ib}^b(t) \\ &\approx \frac{1}{2} (\mathbf{C}_b^e(-) + \mathbf{C}_b^e(+)) \mathbf{f}_{ib}^b\end{aligned}\quad (5.21)$$

or

$$\begin{aligned}\mathbf{v}_{ib}^e &= \overline{\mathbf{C}}_b^e \mathbf{v}_{ib}^b \\ &\approx \frac{1}{2} (\mathbf{C}_b^e(-) + \mathbf{C}_b^e(+)) \mathbf{v}_{ib}^b\end{aligned}\quad (5.22)$$

5.2.3 Velocity Update

As in the inertial-frame implementation, the reference frame and resolving axes are the same, so, from (2.44) and (2.45),

$$\dot{\mathbf{v}}_{eb}^e = \mathbf{a}_{eb}^e = \ddot{\mathbf{r}}_{eb}^e \quad (5.23)$$

Now, applying (2.30), (2.33), and (2.34) in turn,

$$\begin{aligned}\mathbf{r}_{eb}^e &= \mathbf{r}_{ib}^e - \mathbf{r}_{ie}^e \\ &= \mathbf{r}_{ib}^e\end{aligned}\quad (5.24)$$

Substituting this into (5.23),

$$\dot{\mathbf{v}}_{eb}^e = \ddot{\mathbf{r}}_{ib}^e \quad (5.25)$$

Applying (2.49), noting that the Earth rate, $\boldsymbol{\omega}_{ie}^e$, is constant,

$$\dot{\mathbf{v}}_{eb}^e = -\boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{ib}^e - 2\boldsymbol{\Omega}_{ie}^e \dot{\mathbf{r}}_{eb}^e + \mathbf{a}_{ib}^e \quad (5.26)$$

Thus, the rate of change of velocity resolved about the Earth-frame axes incorporates a centrifugal and a Coriolis term due to the rotation of the resolving axes. Applying (2.34) and (2.35),

$$\dot{\mathbf{v}}_{eb}^e = -\boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e + \mathbf{a}_{ib}^e \quad (5.27)$$

From (2.76), the applied acceleration, \mathbf{a}_{ib}^e , is the sum of the measured specific force, \mathbf{f}_{ib}^e , and the acceleration due to the gravitational force, \mathbf{g}_b^e . From (2.83), the acceleration due to gravity, \mathbf{g}_b^e , is the sum of the gravitational and centrifugal accelerations. Substituting these into (5.27),

$$\dot{\mathbf{v}}_{eb}^e = \mathbf{f}_{ib}^e + \mathbf{g}_b^e(\mathbf{r}_{eb}^e) - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e \quad (5.28)$$

An analytical solution is complex. However, as the Coriolis term will be much smaller than the specific-force and gravity terms, except for space applications, it is a reasonable approximation to neglect the variation of the Coriolis term over the integration interval. Thus,

$$\begin{aligned} \mathbf{v}_{eb}^e(+) &\approx \mathbf{v}_{eb}^e(-) + (\mathbf{f}_{ib}^e + \mathbf{g}_b^e(\mathbf{r}_{eb}^e(-)) - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e(-)) \tau_i \\ &= \mathbf{v}_{eb}^e(-) + \mathbf{v}_{ib}^e + (\mathbf{g}_b^e(\mathbf{r}_{eb}^e(-)) - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e(-)) \tau_i \end{aligned} \quad (5.29)$$

Most gravity models operate as a function of latitude and height, calculated from Cartesian ECEF position using (2.71). The gravity is converted from local navigation frame to ECEF resolving axes by premultiplying by \mathbf{C}_n^e , given by (2.99). Alternatively, a gravity model formulated in ECEF axes is presented in [3].

5.2.4 Position Update

In the ECEF-frame navigation equations, the reference and resolving frames are the same, so, from (2.36),

$$\dot{\mathbf{r}}_{eb}^e = \mathbf{v}_{eb}^e \quad (5.30)$$

Integrating this, assuming the velocity varies linearly over the integration interval,

$$\begin{aligned} \mathbf{r}_{eb}^e(+) &= \mathbf{r}_{eb}^e(-) + (\mathbf{v}_{eb}^e(-) + \mathbf{v}_{eb}^e(+)) \frac{\tau_i}{2} \\ &\approx \mathbf{r}_{eb}^e(-) + \mathbf{v}_{eb}^e(-) \tau_i + (\mathbf{f}_{ib}^e + \mathbf{g}_b^e(\mathbf{r}_{eb}^e(-)) - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e(-)) \frac{\tau_i^2}{2} \end{aligned} \quad (5.31)$$

5.3 Local-Navigation-Frame Navigation Equations

In the local-navigation-frame implementation of the inertial navigation equations, the ECEF frame is used as the reference frame, while the local navigation frame (north, east, down) comprises the resolving axes. Thus, attitude is expressed as the body-to-navigation-frame coordinate transformation matrix, \mathbf{C}_b^n , and velocity is Earth-referenced in local navigation frame axes, \mathbf{v}_{eb}^n . Position is expressed in the curvilinear form (i.e., as geodetic latitude, L_b , longitude, λ_b , and geodetic height,

b_b) and is commonly integrated directly from the velocity rather than converted from its Cartesian form.

This form of navigation equations has the advantage of providing a navigation solution in a form readily suited for user output. However, additional complexity is introduced, compared to the ECI and ECEF-frame implementations, as the orientation of the resolving axes with respect to the reference frame depends on the position. Figure 5.4 is a block diagram showing how the angular-rate and specific-force measurements are used to update the attitude, velocity, and position in a local-navigation-frame implementation. Each of the four steps is described in turn. This is followed by a brief discussion of the related wander-azimuth implementation.

5.3.1 Attitude Update

The attitude update step of the local-navigation-frame navigation equations uses the position and velocity solution as well as the angular-rate measurement to update \mathbf{C}_b^n . This is necessary because the orientation of the north, east, and down axes changes as the navigation system moves with respect to the Earth, as explained in Section 2.1.3. From (2.28), the time derivative of the coordinate transformation matrix is

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \boldsymbol{\Omega}_{nb}^b \quad (5.32)$$

Using (2.25) and (2.27), this may be split into three terms:

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \boldsymbol{\Omega}_{ib}^b - (\boldsymbol{\Omega}_{ie}^n + \boldsymbol{\Omega}_{en}^n) \mathbf{C}_b^n \quad (5.33)$$

The first term is due to the inertially referenced angular rate, measured by the gyros, and the second is due to the rotation of the Earth with respect to an inertial

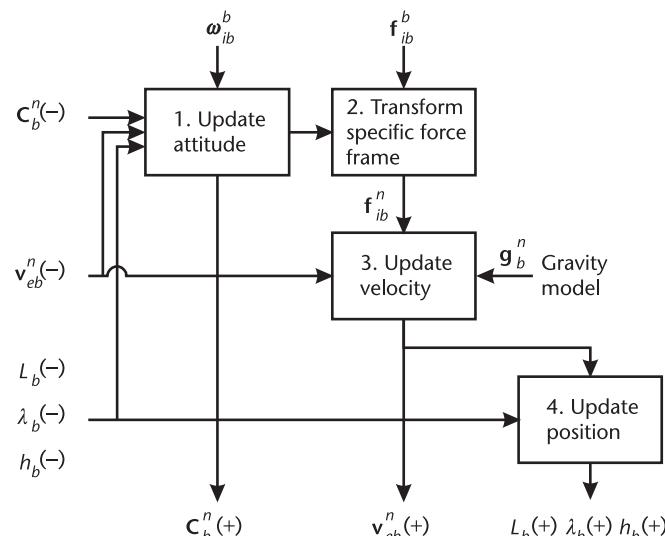


Figure 5.4 Block diagram of local-navigation-frame navigation equations.

frame. The third term, known as the *transport rate*, arises from the rotation of the local-navigation-frame axes as the frame center (i.e., the navigation system) moves with respect to the Earth. When the attitude of the body frame with respect to the local navigation frame remains constant, the gyros sense the Earth rotation and transport rate, which must be corrected for to keep the attitude unchanged.

The Earth-rotation vector in local navigation frame axes is given by (2.75), so the skew-symmetric matrix is

$$\boldsymbol{\Omega}_{ie}^n = \omega_{ie} \begin{pmatrix} 0 & \sin L_b & 0 \\ -\sin L_b & 0 & -\cos L_b \\ 0 & \cos L_b & 0 \end{pmatrix} \quad (5.34)$$

noting that this is a function of latitude.

From (2.28), the transport rate may be obtained by solving,

$$\dot{\mathbf{C}}_e^n = \boldsymbol{\Omega}_{en}^n \mathbf{C}_e^n \quad (5.35)$$

The ECEF-to-local-navigation-frame coordinate transformation matrix is given by (2.99). Taking the time derivative of this gives

$$\dot{\mathbf{C}}_e^n = \left[\begin{pmatrix} -\dot{\lambda}_b \cos L_b \\ \dot{L}_b \\ \dot{\lambda}_b \sin L_b \end{pmatrix} \wedge \right] \mathbf{C}_e^n \quad (5.36)$$

Substituting this into (5.35), together with the derivatives of the latitude and longitude from (2.72) gives

$$\boldsymbol{\Omega}_{en}^n = \begin{pmatrix} 0 & -\omega_{en,z}^n & \omega_{en,y}^n \\ \omega_{en,z}^n & 0 & -\omega_{en,x}^n \\ -\omega_{en,y}^n & \omega_{en,x}^n & 0 \end{pmatrix} \quad (5.37)$$

$$\boldsymbol{\omega}_{en}^n = \begin{pmatrix} v_{eb,E}^n / (R_E(L_b) + h_b) \\ -v_{eb,N}^n / (R_N(L_b) + h_b) \\ -v_{eb,E}^n \tan L_b / (R_E(L_b) + h_b) \end{pmatrix}$$

Integrating (5.33) gives

$$\mathbf{C}_b^n(t + \tau) = \mathbf{C}_b^n(t) \exp([\boldsymbol{\alpha}_{ib}^b \wedge]) - \{\exp[(\boldsymbol{\Omega}_{ie}^n + \boldsymbol{\Omega}_{en}^n) \tau_i] - \mathbf{I}_3\} \mathbf{C}_b^n(t) \quad (5.38)$$

Obtaining a full analytical solution to this is complex, and accounting for the variation of position and velocity over the attitude update interval can require

recursive navigation equations. However, a reasonable approximation for most applications can be obtained by neglecting this variation and truncating the power-series expansion of the exponential terms to first order, giving

$$\mathbf{C}_b^n(+) \approx \mathbf{C}_b^n(-) (\mathbf{I}_3 + \boldsymbol{\Omega}_{ib}^b \tau_i) - (\boldsymbol{\Omega}_{ie}^n(-) + \boldsymbol{\Omega}_{en}^n(-)) \mathbf{C}_b^n(-) \tau_i \quad (5.39)$$

where $\boldsymbol{\Omega}_{ie}^n(-)$ is calculated using $L_b(-)$ and $\boldsymbol{\Omega}_{en}^n(-)$ is calculated using $L_b(-)$, $h_b(-)$, and $\mathbf{v}_{eb}^n(-)$. Higher precision solutions are discussed in Section 5.4.2.

5.3.2 Specific-Force Frame Transformation

The specific-force frame transformation is essentially the same as for the ECI and ECEF-frame implementations. Thus,

$$\begin{aligned} \mathbf{f}_{ib}^n(t) &= \mathbf{C}_b^n(t) \mathbf{f}_{ib}^b(t) \\ &\approx \frac{1}{2} (\mathbf{C}_b^n(-) + \mathbf{C}_b^n(+)) \mathbf{f}_{ib}^b \end{aligned} \quad (5.40)$$

or

$$\begin{aligned} \mathbf{v}_{ib}^n &= \bar{\mathbf{C}}_b^n \mathbf{v}_{ib}^b \\ &\approx \frac{1}{2} (\mathbf{C}_b^n(-) + \mathbf{C}_b^n(+)) \mathbf{v}_{ib}^b \end{aligned} \quad (5.41)$$

The accuracy of this approximation will be similar to that in the inertial frame as the gyro-sensed rotation will usually be much larger than the Earth rate and transport rate components.³

5.3.3 Velocity Update

In the local-navigation-frame navigation equations, the resolving axes of the velocity are not the same as its reference frame. From (2.41), the velocity is expressed in terms in terms of its counterpart in ECEF resolving axes by

$$\mathbf{v}_{eb}^n = \mathbf{C}_e^n \mathbf{v}_{eb}^e \quad (5.42)$$

Differentiating this,

$$\dot{\mathbf{v}}_{eb}^n = \dot{\mathbf{C}}_e^n \mathbf{v}_{eb}^e + \mathbf{C}_e^n \dot{\mathbf{v}}_{eb}^e \quad (5.43)$$

Thus, there is a transport-rate term in addition to the applied acceleration, centrifugal, and Coriolis terms found in the ECEF frame velocity update described in Section 5.2.3. Applying (2.28) and (2.41) to the first term and substituting (5.27) for the second term,

$$\dot{\mathbf{v}}_{eb}^n = -\boldsymbol{\Omega}_{en}^n \mathbf{v}_{eb}^n + \mathbf{C}_e^n (-\boldsymbol{\Omega}_{ie}^e \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{eb}^e - 2\boldsymbol{\Omega}_{ie}^e \mathbf{v}_{eb}^e + \mathbf{a}_{ib}^e) \quad (5.44)$$

Applying (2.27), (2.31), (2.41), and (2.51) to transform the resolving axes and rearranging gives

$$\dot{\mathbf{v}}_{eb}^n = -\boldsymbol{\Omega}_{ie}^n \boldsymbol{\Omega}_{ie}^n \mathbf{r}_{eb}^n - (\boldsymbol{\Omega}_{en}^n + 2\boldsymbol{\Omega}_{ie}^n) \mathbf{v}_{eb}^n + \mathbf{a}_{ib}^n \quad (5.45)$$

noting that the skew-symmetric matrices of the Earth rotation and transport rate are given by (5.34) and (5.37), respectively.

Expressing the acceleration in terms of the specific force, gravity, and centrifugal acceleration using (2.76) and (2.83) gives

$$\dot{\mathbf{v}}_{eb}^n = \mathbf{f}_{ib}^n + \mathbf{g}_b^n(L_b, h_b) - (\boldsymbol{\Omega}_{en}^n + 2\boldsymbol{\Omega}_{ie}^n) \mathbf{v}_{eb}^n \quad (5.46)$$

where the acceleration due to gravity is modeled as a function of latitude and height. Again, obtaining a full analytical solution is complex. However, as the Coriolis and transport-rate terms will generally be the smallest, it is a reasonable approximation to neglect their variation over the integration interval. Again, the variation of the acceleration due to gravity over the integration interval can generally be neglected. Thus,

$$\begin{aligned} \mathbf{v}_{eb}^n(+) &\approx \mathbf{v}_{eb}^n(-) + [\mathbf{f}_{ib}^n + \mathbf{g}_b^n(L_b(-), h_b(-)) - (\boldsymbol{\Omega}_{en}^n(-) + 2\boldsymbol{\Omega}_{ie}^n(-)) \mathbf{v}_{eb}^n(-)] \tau_i \quad (5.47) \\ &= \mathbf{v}_{eb}^n(-) + \mathbf{v}_{ib}^n + [\mathbf{g}_b^n(L_b(-), h_b(-)) - (\boldsymbol{\Omega}_{en}^n(-) + 2\boldsymbol{\Omega}_{ie}^n(-)) \mathbf{v}_{eb}^n(-)] \tau_i \end{aligned}$$

5.3.4 Position Update

From (2.72), the derivatives of the latitude, longitude, and height are functions of the velocity, latitude, and height. Thus,¹

$$\begin{aligned} L_b(+) &= L_b(-) + \int_t^{t+\tau_i} \frac{v_{eb,N}^n(t')}{R_N(L_b(t')) + h_b(t')} dt' \\ \lambda_b(+) &= \lambda_b(-) + \int_t^{t+\tau_i} \frac{v_{eb,E}^n(t')}{(R_E(L_b(t')) + h_b(t')) \cos L_b(t')} dt' \quad (5.48) \\ h_b(+) &= h_b(-) - \int_t^{t+\tau_i} v_{eb,D}^n(t') dt' \end{aligned}$$

The variation of the meridian and transverse radii of curvature, R_N and R_E , with the geodetic latitude, L_b , is weak, so it is acceptable to neglect their variation with latitude over the integration interval. Assuming the velocity varies as a linear

function of time over the integration interval, a suitable approximation for the position update is

$$\begin{aligned} h_b(+) &= h_b(-) - \frac{\tau_i}{2} (v_{eb,D}^n(-) + v_{eb,D}^n(+)) \\ L_b(+) &\approx L_b(-) + \frac{\tau_i}{2} \left(\frac{v_{eb,N}^n(-)}{R_N(L_b(-)) + h_b(-)} + \frac{v_{eb,N}^n(+)}{R_N(L_b(-)) + h_b(+)} \right) \\ \lambda_b(+) &= \lambda_b(-) + \frac{\tau_i}{2} \left(\frac{v_{eb,E}^n(-)}{(R_E(L_b(-)) + h_b(-)) \cos L_b(-)} + \frac{v_{eb,E}^n(+)}{(R_E(L_b(+)) + h_b(+)) \cos L_b(+)} \right) \end{aligned} \quad (5.49)$$

noting that the height, latitude, and longitude should be calculated in that order.²

5.3.5 Wander-Azimuth Implementation

The wander-azimuth coordinate frame (Section 2.1.5), denoted w , is closely related to the local navigation frame. The z axis is coincidental, pointing down, but the x and y axes are rotated about the z axis with respect to the local navigation frame by a wander angle that varies with position. The wander angle is simply the heading (or azimuthal) Euler angle from the local navigation frame to the wander azimuth frame, ψ_{nw} , though many authors use α . Thus, from (2.14) and (2.15),

$$\mathbf{C}_n^w = \begin{pmatrix} \cos \psi_{nw} & \sin \psi_{nw} & 0 \\ -\sin \psi_{nw} & \cos \psi_{nw} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \mathbf{C}_w^n = \begin{pmatrix} \cos \psi_{nw} & -\sin \psi_{nw} & 0 \\ \sin \psi_{nw} & \cos \psi_{nw} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (5.50)$$

Note that some authors use the wander angle with the opposing sign, ψ_{wn} , which may also be denoted α .

Inertial navigation equations can be mechanized in the wander-azimuth axes to avoid the polar singularities that occur in the local navigation frame. The inertial navigation equations in the wander azimuth frame are as those for the local navigation frame, presented earlier, with w substituted for n , except that the transport-rate term has no component about the vertical axis. Thus,³

$$\boldsymbol{\omega}_{ew}^w = \mathbf{C}_n^w \begin{pmatrix} \omega_{en,N}^n \\ \omega_{en,E}^n \\ 0 \end{pmatrix} \quad (5.51)$$

The Earth-rotation vector is, from (2.92),

$$\boldsymbol{\omega}_{ie}^w = \mathbf{C}_n^w \boldsymbol{\omega}_{ie}^n \quad (5.52)$$

To calculate the transport rate using (5.37) and (5.51) and to update the latitude, longitude, and height by integrating the velocity as shown in Section 5.3.4, the velocity must first be transformed into local-navigation-frame resolving axes using

$$\mathbf{v}_{eb}^n = \mathbf{C}_w^n \mathbf{v}_{eb}^w \quad (5.53)$$

The wander angle is generally initialized at zero at the start of navigation. Its rate of change may be obtained by calculating $\dot{\mathbf{C}}_w^n$ and $\dot{\mathbf{C}}_w^w$ using (5.33) and substituting in (5.50) to (5.53). This gives

$$\dot{\mathbf{C}}_w^n = \begin{pmatrix} 0 & \omega_{en,D}^n & 0 \\ -\omega_{en,D}^n & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{C}_w^n \quad (5.54)$$

Substituting in (5.50) and integrating over the navigation equations update interval gives the wander angle update equation:

$$\begin{aligned} \psi_{nw}(+) &= \psi_{nw}(-) - \omega_{en,D}^n \tau_i \\ &= \psi_{nw}(-) + \frac{v_{eb,E}^n \tan L_b \tau_i}{R_E(L_b) + h_b} \\ &= \psi_{nw}(-) + \dot{\lambda}_b \tan L_b \tau_i \end{aligned} \quad (5.55)$$

5.4 Navigation Equations Precision

The design of a set of inertial navigation equations is a tradeoff between accuracy, processing efficiency, and complexity. It is possible to optimize two of these, but not all three. The accuracy of the navigation equations is a function of three factors: the iteration rate, the nature of the approximations made, and the dynamic and vibration environment. This section explores each of these factors.

In determining the accuracy requirements for a set of inertial navigation equations, it is important to consider the navigation system as a whole. For example, where the inertial sensors are relatively poor, an improvement in the accuracy of the navigation equations may have negligible impact on overall performance. Another consideration is the degree to which integration with other navigation sensors can correct the errors of the INS [4].

Traditionally, the accuracy requirements for inertial navigation have been high, as INS with high-quality inertial sensors have been used for sole-means navigation in the horizontal axes, or with infrequent position updates, for periods of hours. Until the 1990s, processing power was also at a premium. Hence considerable effort was expended developing highly accurate and highly efficient, but also highly complex, navigation algorithms (e.g., [2]). However, today, a faster processor can often be more cost effective than expending a large amount of effort designing,

implementing, and debugging complex navigation equations. Also, where the navigation system incorporates a Kalman filter for integrating multiple sensors, the processing requirements for the inertial navigation equations will generally be a small fraction of the total.

5.4.1 Iteration Rates

In solving the inertial navigation equations in the preceding sections, approximations have been made. Errors also arise from processing the inertial sensor measurements at a lower rate than the IMU outputs them, as this constitutes an approximation that the specific force and angular rate are constant over the averaging interval. At a given level of dynamics, the larger the integration step (i.e., the lower the iteration rate), the larger the errors arising from these approximations will be. Thus, by increasing the iteration rate of a given set of navigation equations, the accuracy can be improved.¹

Different approximations in different stages of the navigation equations have differing impacts on the overall error budget at a given iteration rate. The relative impact will also depend on the dynamics, with different errors varying with different kinematic quantities. For example, attitude integration errors vary with the angular rate. Thus, if the navigation-equations error budget is to be divided roughly equally between different stages, different iteration rates should be used.

A further issue in the ECEF and local-navigation-frame implementations is that the Earth-rate, transport-rate, and Coriolis terms tend to be much smaller than the terms derived from the accelerometer and gyro measurements. Consequently, there is scope to iterate calculation of these terms at a lower rate.

To gain full benefits of iterating an equation at a higher rate, the inputs from the previous stage must be at the same rate (or higher). This can be illustrated with position and velocity. The position increment from time t to time $t + \tau$ is obtained by integrating the velocity. Figure 5.5(a) shows that where the velocity is only known at the beginning and end of the integration interval, dividing the integral into two trapezoids does not change the result. Figure 5.5(b) shows that where the velocity is also known at the halfway point, dividing the position integral into two trapezoids does improve precision.

Increasing the rate of data transmission from one navigation equations stage to the next can be done in two ways. Either intermediate outputs from the first

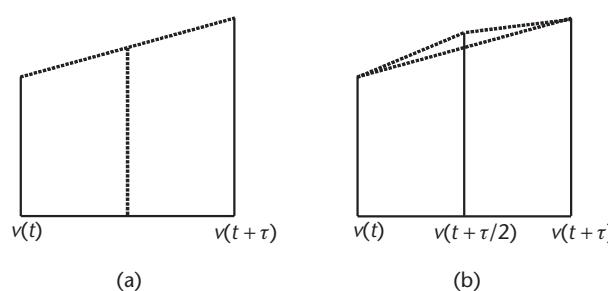


Figure 5.5 (a, b) Integration of velocity to obtain position. (From: [5]. © 2002 QinetiQ Ltd. Reprinted with permission.)

step may be passed onto the second step or the navigation equations function can step between the first and second (and further) stages at the higher rate. However, where the local-navigation-frame equations are used (Section 5.3), the attitude and velocity integrations are a function of position. Therefore, stepping through the complete navigation equations cycle at a higher rate will generally be more accurate than passing intermediate outputs from one step to the next.²

Where exact navigation equations are used, there is no benefit in processing them at a higher iteration rate than the IMU output rate. However, where approximations are made, some stages, such as the attitude update and specific-force frame transformation, can benefit from a higher iteration rate. An alternative is to implement higher order numerical integration techniques. For example, a fourth-order Runge-Kutta algorithm integrates

$$\mathbf{x}(t + \tau) = \mathbf{x}(t) + \int_t^{t + \tau} \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) dt \quad (5.56)$$

in five steps:

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{k}_2 &= \mathbf{g}\left(\left(\mathbf{x}(t) + \frac{1}{2} \tau \mathbf{k}_1\right), \mathbf{u}\left(t + \frac{1}{2} \tau\right)\right) \\ \mathbf{k}_3 &= \mathbf{g}\left(\left(\mathbf{x}(t) + \frac{1}{2} \tau \mathbf{k}_2\right), \mathbf{u}\left(t + \frac{1}{2} \tau\right)\right) \\ \mathbf{k}_4 &= \mathbf{g}((\mathbf{x}(t) + \tau \mathbf{k}_3), \mathbf{u}(t + \tau)) \\ \mathbf{x}(t + \tau) &= \mathbf{x}(t) + \frac{1}{6} \tau (\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4) \end{aligned} \quad (5.57)$$

The Runge-Kutta approach may be applied to individual stages of the navigation equations or to the processing cycle as a whole.

5.4.2 Attitude Update

It is convenient to define the attitude update matrix as the coordinate transformation matrix from the body frame at the end of the attitude update step of the navigation equations to that at the beginning, \mathbf{C}_{b+}^{b-} (some authors use \mathbf{A}). It may be used to define the attitude update step in the ECI frame; thus,

$$\begin{aligned} \mathbf{C}_b^i (+) &= \mathbf{C}_b^i (-) \mathbf{C}_{b+}^{b-} \\ \mathbf{C}_{b+}^{b-} &= \mathbf{C}_i^b (-) \mathbf{C}_b^i (+) \end{aligned} \quad (5.58)$$

Substituting (5.3) and (5.4) into (5.58) defines the attitude update matrix in terms of the attitude increment, $\boldsymbol{\alpha}_{ib}^b$:

$$\mathbf{C}_{b+}^{b-} = \mathbf{C}_{b(t+\tau_i)}^{b(t)} = \exp[\boldsymbol{\alpha}_{ib}^b \wedge] = \sum_{r=0}^{\infty} \frac{[\boldsymbol{\alpha}_{ib}^b \wedge]^r}{r!} \quad (5.59)$$

Where the power-series expansion is truncated, errors arise depending on the step size of the attitude increment and the order at which the power series is truncated. Table 5.1 presents some examples. Clearly, the third- and fourth-order algorithms perform significantly better than the first- and second-order algorithms. It should also be noted that the error varies as the square of the attitude increment for the first- and second-order algorithms, but as the fourth power for the third- and fourth-order variants. Thus, with the higher order algorithms, increasing the iteration rate has more impact on the accuracy.

The third and fourth powers of a skew-symmetric matrix have the following properties:

$$\begin{aligned} [\mathbf{x} \wedge]^3 &= -|\mathbf{x}|^2 [\mathbf{x} \wedge] \\ [\mathbf{x} \wedge]^4 &= -|\mathbf{x}|^2 [\mathbf{x} \wedge]^2 \end{aligned} \quad (5.60)$$

Substituting this into (5.4):

$$\mathbf{C}_{b+}^{b-} = \mathbf{I}_3 + \left(\sum_{r=0}^{\infty} (-1)^r \frac{|\boldsymbol{\alpha}_{ib}^b|^{2r}}{(2r+1)!} \right) [\boldsymbol{\alpha}_{ib}^b \wedge] + \left(\sum_{r=0}^{\infty} (-1)^r \frac{|\boldsymbol{\alpha}_{ib}^b|^{2r}}{(2r+2)!} \right) [\boldsymbol{\alpha}_{ib}^b \wedge]^2 \quad (5.61)$$

The fourth-order approximation is then

$$\mathbf{C}_{b+}^{b-} \approx \mathbf{I}_3 + \left(1 - \frac{|\boldsymbol{\alpha}_{ib}^b|^2}{6} \right) [\boldsymbol{\alpha}_{ib}^b \wedge] + \left(1 - \frac{|\boldsymbol{\alpha}_{ib}^b|^2}{24} \right) [\boldsymbol{\alpha}_{ib}^b \wedge]^2 \quad (5.62)$$

However, the power-series expansions in (5.61) are closely related to those of the sine and cosine, so

$$\mathbf{C}_{b+}^{b-} = \mathbf{I}_3 + \frac{\sin |\boldsymbol{\alpha}_{ib}^b|}{|\boldsymbol{\alpha}_{ib}^b|} [\boldsymbol{\alpha}_{ib}^b \wedge] + \frac{1 - \cos |\boldsymbol{\alpha}_{ib}^b|}{|\boldsymbol{\alpha}_{ib}^b|^2} [\boldsymbol{\alpha}_{ib}^b \wedge]^2 \quad (5.63)$$

Table 5.1 Drift of First- to Fourth-Order Attitude Update Algorithms

Algorithm Order	Attitude Drift, rad s ⁻¹ (° hr ⁻¹) $\boldsymbol{\alpha}$ = 0.1 rad step size	Attitude Drift, rad s ⁻¹ (° hr ⁻¹) $\boldsymbol{\alpha}$ = 0.05 rad step size
1	0.033 (6870)	8.3×10^{-3} (1720)
2	0.017 (3430)	4.2×10^{-3} (860)
3	3.4×10^{-5} (69)	2.5×10^{-6} (0.4)
4	8.2×10^{-6} (1.7)	5.2×10^{-7} (0.1)

The ECI-frame attitude update may thus be performed exactly. Note that the inverse of (5.63) gives the attitude increment vector in terms of the attitude update matrix:

$$\boldsymbol{\alpha}_{ib}^b = \frac{\theta}{2 \sin \theta} \begin{pmatrix} C_{b+3,2}^{b-} - C_{b+2,3}^{b-} \\ C_{b+1,3}^{b-} - C_{b+3,1}^{b-} \\ C_{b+2,1}^{b-} - C_{b+1,2}^{b-} \end{pmatrix}, \quad \theta = \arccos \left[\frac{\text{tr}(\mathbf{C}_{b+}^{b-}) - 1}{2} \right] \quad (5.64)$$

A similar approach may be taken with the ECEF-frame attitude update. For precision, the first-order solution, (5.20), is replaced by

$$\mathbf{C}_b^e(+) = \mathbf{C}_b^e(-) \mathbf{C}_{b+}^{b-} - \boldsymbol{\Omega}_{ie}^e \mathbf{C}_b^e(-) \tau \quad (5.65)$$

where the attitude update matrix is given by (5.63) as before. Note that the first-order approximation has been retained for the Earth-rate term, as this introduces an error of only 2.7×10^{-10} rad s $^{-1}$ (5.5×10^{-5} ° hr $^{-1}$) at a 10-Hz update rate and 2.7×10^{-11} rad s $^{-1}$ (5.5×10^{-6} ° hr $^{-1}$) at a 100-Hz update rate, which is much less than the bias of even the most accurate gyros. Thus, this is an exact solution for all practical purposes.

In the local-navigation-frame attitude update, there is also a transport-rate term, $\boldsymbol{\omega}_{en}^n$, given by (5.37). For velocities up to 467 m s $^{-1}$ (Mach 1.4), this is less than the Earth-rotation rate, so for the vast majority of applications, it is valid to truncate the power-series expansion of $\exp(\boldsymbol{\Omega}_{en}^n \tau)$ to first order. Thus, for improved precision, the first-order solution, (5.39), is replaced by

$$\mathbf{C}_b^n(+) \approx \mathbf{C}_b^n(-) \mathbf{C}_{b+}^{b-} - (\boldsymbol{\Omega}_{ie}^n(-) + \boldsymbol{\Omega}_{en}^n(-)) \mathbf{C}_b^n(-) \tau \quad (5.66)$$

However, for high-precision, high-dynamic applications, the variation of the transport rate over the update interval can be significant. Where a high-precision specific-force frame transformation (Section 5.4.3) is implemented, the updated attitude is not required at that stage. This enables the attitude update step to be moved from the beginning to the end of the navigation equations processing cycle, enabling an averaged transport rate to be used for the attitude update:

$$\mathbf{C}_b^n(+) = \mathbf{C}_b^n(-) \mathbf{C}_{b+}^{b-} - \left(\boldsymbol{\Omega}_{ie}^n(-) + \frac{1}{2} \boldsymbol{\Omega}_{en}^n(-) + \frac{1}{2} \boldsymbol{\Omega}_{en}^n(+) \right) \mathbf{C}_b^n(-) \tau \quad (5.67)$$

where $\boldsymbol{\Omega}_{en}^n(+)$ is calculated using $L_b(+)$, $h_b(+)$, and $\mathbf{v}_{eb}^n(+)$. Figure 5.6 shows the modified block diagram for the precision local-navigation-frame navigation equations.

To reduce the processing load, the attitude update is sometimes iterated at a lower rate than the IMU output rate. Where the integration interval for the attitude update comprises n IMU output intervals, the attitude update matrix becomes

$$\mathbf{C}_{b+}^{b-} = \exp[\boldsymbol{\alpha}_{ib,1}^b \wedge] \exp[\boldsymbol{\alpha}_{ib,2}^b \wedge] \dots \exp[\boldsymbol{\alpha}_{ib,n}^b \wedge] \quad (5.68)$$

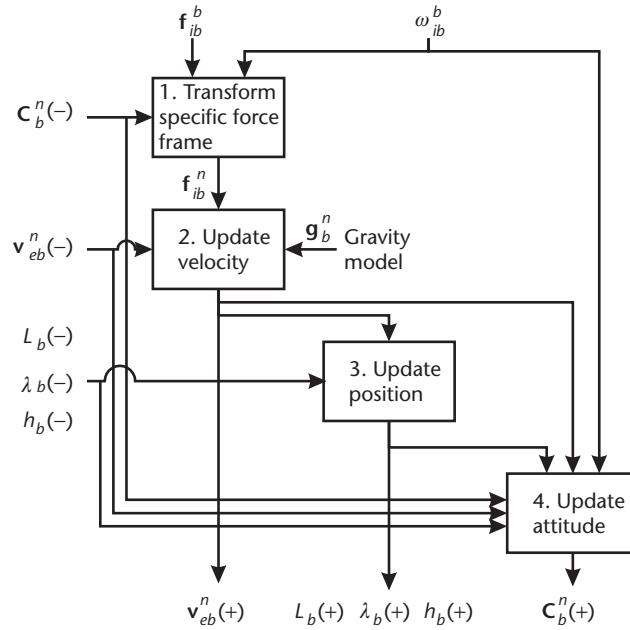


Figure 5.6 Block diagram of precision local-navigation-frame navigation equations.

where

$$\alpha_{ib,j}^b = \int_{t+(j-1)\tau/n}^{t+j\tau/n} \omega_{ib}^b(t') dt' \quad (5.69)$$

Implementing (5.68) as it stands offers no computational saving over performing the attitude update at the IMU rate. From (2.23), (5.63), and (5.59), the attitude update matrix may be expressed in terms of a rotation vector [6]:

$$C_{b+}^{b-} = \exp[\rho_{b-b+} \wedge] \quad (5.70)$$

Note that the body-frame rotation vector, ρ_{b-b+} , is equal to the attitude increment of the body frame with respect to inertial space in body-frame axes, α_{ib}^b , over the same time interval. Thus,

$$\rho_{b-b+} = \int_t^{t+\tau} \omega_{ib}^b(t') dt' \quad (5.71)$$

Note, however, that rotation vectors and attitude increments are not the same in general.

As the direction of rotation varies between successive measurements, the rotation vector is not simply the sum of the attitude increments. In physical terms, this is because the resolving axes vary between successive attitude increments. In

mathematical terms, the skew-symmetric matrices of successive attitude increments do not commute.

From [6], the rate of change of the rotation vector varies with the angular rate as

$$\begin{aligned}\dot{\boldsymbol{\rho}}_{b-b+} &= \boldsymbol{\omega}_{ib}^b + \frac{1}{2} \boldsymbol{\rho}_{b-b+} \wedge \boldsymbol{\omega}_{ib}^b \\ &\quad + \frac{1}{|\boldsymbol{\rho}_{b-b+}|^2} \left[1 - \frac{|\boldsymbol{\rho}_{b-b+}| \sin |\boldsymbol{\rho}_{b-b+}|}{2(1 - \cos |\boldsymbol{\rho}_{b-b+}|)} \right] \boldsymbol{\rho}_{b-b+} \wedge \boldsymbol{\rho}_{b-b+} \wedge \boldsymbol{\omega}_{ib}^b\end{aligned}\quad (5.72)$$

From [2, 7], a second-order approximation incorporating only the first two terms of (5.72) gives the following solution:

$$\boldsymbol{\rho}_{b-b+} \approx \sum_{j=1}^n \boldsymbol{\alpha}_{ib,j}^b + \frac{1}{2} \sum_{j=1}^{n-1} \sum_{k=j+1}^n \boldsymbol{\alpha}_{ib,j}^b \wedge \boldsymbol{\alpha}_{ib,k}^b \quad (5.73)$$

Note that, where sufficient processing capacity is available, it is both simpler and more accurate to iterate the attitude update at the IMU output rate.

Coordinate transformation matrices are orthonormal, (2.13), so the scalar product of any two rows or any two columns should be zero. Orthonormality is maintained through exact navigation equations. However, the use of approximations and the presence of computational rounding errors can cause departures from this. Consequently, it can be useful to implement a reorthogonalization and renormalization algorithm at regular intervals.¹

Breaking down the coordinate transformation matrix (frames omitted) into three rows,

$$\mathbf{C} = \begin{pmatrix} \mathbf{c}_1^T \\ \mathbf{c}_2^T \\ \mathbf{c}_3^T \end{pmatrix} \quad (5.74)$$

Orthogonalization is achieved by calculating $\Delta_{ij} = \mathbf{c}_i^T \mathbf{c}_j$ for each pair of rows and apportioning a correction equally between them:

$$\begin{aligned}\mathbf{c}_1(+) &= \mathbf{c}_1(-) - \frac{1}{2} \Delta_{12} \mathbf{c}_2(-) - \frac{1}{2} \Delta_{13} \mathbf{c}_3(-) \\ \mathbf{c}_2(+) &= \mathbf{c}_2(-) - \frac{1}{2} \Delta_{12} \mathbf{c}_1(-) - \frac{1}{2} \Delta_{23} \mathbf{c}_3(-) \\ \mathbf{c}_3(+) &= \mathbf{c}_3(-) - \frac{1}{2} \Delta_{13} \mathbf{c}_1(-) - \frac{1}{2} \Delta_{23} \mathbf{c}_2(-)\end{aligned}\quad (5.75)$$

Normalization is applied to each row by

$$\mathbf{c}_i(+) = \frac{2}{1 + \mathbf{c}_i^T(-)\mathbf{c}_i(-)} \mathbf{c}_i(-) \quad (5.76)$$

The orthonormalization may also be performed by the column.

5.4.3 Specific-Force Frame Transformation

The specific force in ECI-frame resolving axes is instantaneously related to that in the body-frame axes by repeating (5.8):

$$\mathbf{f}_{ib}^i(t) = \mathbf{C}_b^i(t) \mathbf{f}_{ib}^b(t)$$

The IMU outputs the average specific force over the interval t to $t + \tau_i$, and the ECI-axes specific force is similarly averaged. The transformation is thus

$$\mathbf{f}_{ib}^i = \bar{\mathbf{C}}_b^i \mathbf{f}_{ib}^b \quad (5.77)$$

where the average coordinate transformation matrix over the time interval is

$$\bar{\mathbf{C}}_b^i = \frac{1}{\tau_i} \int_t^{t + \tau_i} \mathbf{C}_b^i(t') dt' \quad (5.78)$$

Substituting in (5.4), noting that the variation of the angular rate over the integration interval is unknown,

$$\begin{aligned} \bar{\mathbf{C}}_b^i &= \frac{1}{\tau_i} \mathbf{C}_b^i(-) \int_0^{t + \tau_i} \sum_{r=0}^{\infty} \frac{\{(t'/\tau_i)[\boldsymbol{\alpha}_{ib}^b \wedge]\}^r}{r!} dt' \\ &= \mathbf{C}_b^i(-) \sum_{r=0}^{\infty} \frac{[\boldsymbol{\alpha}_{ib}^b \wedge]^r}{(r+1)!} \end{aligned} \quad (5.79)$$

Applying (5.60),

$$\begin{aligned} \bar{\mathbf{C}}_b^i &= \mathbf{C}_b^i(-) \mathbf{C}_b^{b-} \\ \mathbf{C}_b^{b-} &= \mathbf{I}_3 + \frac{1 - \cos|\boldsymbol{\alpha}_{ib}^b|}{|\boldsymbol{\alpha}_{ib}^b|^2} [\boldsymbol{\alpha}_{ib}^b \wedge] + \frac{1}{|\boldsymbol{\alpha}_{ib}^b|^2} \left(1 - \frac{\sin|\boldsymbol{\alpha}_{ib}^b|}{|\boldsymbol{\alpha}_{ib}^b|} \right) [\boldsymbol{\alpha}_{ib}^b \wedge]^2 \end{aligned} \quad (5.80)$$

Substituting this into (5.77) or (5.10), the specific force in ECI-frame resolving axes, \mathbf{f}_{ib}^i , or the integrated specific force, \mathbf{v}_{ib}^i , may be calculated exactly.

Retaining the first-order approximation for the Earth-rate term, the precise transformation of the specific force to ECEF-frame axes is

$$\mathbf{f}_{ib}^e = \bar{\mathbf{C}}_b^e \mathbf{f}_{ib}^b, \quad \bar{\mathbf{C}}_b^e = \mathbf{C}_b^e(-) \mathbf{C}_b^{b-} - \frac{1}{2} \boldsymbol{\Omega}_{ie}^e \mathbf{C}_b^e(-) \tau_i \quad (5.81)$$

To transform the specific force to local-navigation-frame axes, the first order approximation is also used for the transport-rate term, as the velocity at time $t + \tau_i$ has yet to be computed:

$$\mathbf{f}_{ib}^n = \bar{\mathbf{C}}_b^n \mathbf{f}_{ib}^b, \quad \bar{\mathbf{C}}_b^n = \mathbf{C}_b^n(-) \mathbf{C}_b^{b-} - \frac{1}{2} (\boldsymbol{\Omega}_{ie}^n(-) + \boldsymbol{\Omega}_{en}^n(-)) \mathbf{C}_b^n(-) \tau_i \quad (5.82)$$

To transform integrated specific force to ECEF and local-navigation-frame axes, $\bar{\mathbf{C}}_b^e$ and $\bar{\mathbf{C}}_b^n$ are substituted into (5.22) and (5.41), respectively.

Where the specific force in the resolving axes used for the velocity is integrated over more than one IMU output interval, in order to reduce the processor load, the specific-force transformation should account for the fact that each successive IMU specific-force measurement may be resolved about a different set of axes as the body-frame orientation changes. A second-order transformation and summation of n successive IMU specific force measurements into the ECI frame is, from [1, 2, 7],

$$\mathbf{v}_{ib,\Sigma}^i \approx \mathbf{C}_b^i(-) \left[\sum_{j=1}^n \mathbf{v}_{ib,j}^b + \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \boldsymbol{\alpha}_{ib,j}^b \wedge \mathbf{v}_{ib,k}^b + \frac{1}{2} \sum_{j=1}^{n-1} \sum_{k=j+1}^n (\boldsymbol{\alpha}_{ib,j}^b \wedge \mathbf{v}_{ib,k}^b - \boldsymbol{\alpha}_{ib,k}^b \wedge \mathbf{v}_{ib,j}^b) \right] \quad (5.83)$$

where $\mathbf{v}_{ib,j}^b$ and $\boldsymbol{\alpha}_{ib,j}^b$ are the j th integrated-specific-force and attitude-increment outputs from the IMU, and $\mathbf{v}_{ib,\Sigma}^i$ is the summed integrated specific force in ECI resolving axes. Again, where there is sufficient processing capacity, it is simpler and more accurate to iterate the specific-force transformation at the IMU update rate.

5.4.4 Velocity and Position Updates

Where the navigation equations are iterated at the IMU output rate, the ECI-frame velocity and position update equations presented in Sections 5.1.3 and 5.1.4 are exact, except for the variation in gravitation over the update interval, which is small enough to be neglected. However, in the ECEF and local-navigation-frame implementations, exact evaluations of the Coriolis and transport-rate terms requires knowledge of the velocity at the end of the update interval, requiring a recursive solution. For most applications, the first-order approximation in (5.29) and (5.47) is sufficient. However, this may lead to significant errors for high-accuracy, high-dynamic applications. One solution is to predict forward the velocity using previous velocity solutions [2]. A better, but more processor-intensive, solution is a two-step recursive method, shown here for the local-navigation-frame implementation:

$$\mathbf{v}_{eb}^{n'} = \mathbf{v}_{eb}^n(-) + [\mathbf{f}_{ib}^n + \mathbf{g}_b^n(L_b(-), b_b(-)) - (\boldsymbol{\Omega}_{en}^n(-) + 2\boldsymbol{\Omega}_{ie}^n(-))\mathbf{v}_{eb}^n(-)]\tau_i \quad (5.84)$$

$$\mathbf{v}_{eb}^n(+) = \mathbf{v}_{eb}^n(-) + \left\{ \begin{array}{l} \mathbf{f}_{ib}^n + \mathbf{g}_b^n(L_b(-), b_b(-)) - \frac{1}{2} [\boldsymbol{\Omega}_{en}^n(-) + 2\boldsymbol{\Omega}_{ie}^n(-)]\mathbf{v}_{eb}^n(-) \\ - \frac{1}{2} [\boldsymbol{\Omega}_{en}^n(L_b(-), b_b(-), \mathbf{v}_{eb}^{n'}) + 2\boldsymbol{\Omega}_{ie}^n(-)]\mathbf{v}_{eb}^{n'} \end{array} \right\} \tau_i$$

To reduce processor load, the Coriolis and transport-rate terms in the velocity update step may be iterated at a lower rate than the specific-force term. However, the lower the iteration rate, the more likely that a recursive solution will be necessary.

Provided they are iterated at the same rate as the velocity update, the ECI and ECEF frame position updates introduce no further approximations beyond those made in the velocity update, while the effect of the meridian radius of curvature approximation in the local-navigation-frame position update is negligible. Algorithms for improving the accuracy of the position update when the navigation equations are iterated more slowly than the IMU output rate are presented in [2, 7].

Where precision inertial sensors and navigation equations are used, the precision of the gravity model (Section 2.3.5) can significantly affect navigation accuracy [8].

5.4.5 Effects of Vibration

The effects of vibration on inertial navigation performance may be illustrated by the cases of coning and sculling motion. Coning motion is synchronized angular oscillation about two orthogonal axes, as shown in Figure 5.7. If the output of a triad of gyroscopes is integrated over a period, τ , in the presence of coning motion of angular frequency, ω_c , and angular amplitudes, θ_i and θ_j , with a phase difference, ϕ , between the two axes, it can be shown [1] that a false rotation, $\delta\omega_c$, is sensed about the axis orthogonal to θ_i and θ_j , where

$$\delta\omega_c = \omega_c \theta_i \wedge \theta_j \sin \phi \left(1 - \frac{\sin \omega_c \tau}{\omega_c \tau} \right) \quad (5.85)$$

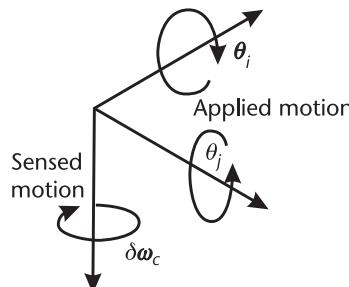


Figure 5.7 Coning motion. (From: [5]. © 2002 QinetiQ Ltd. Reprinted with permission.)

This happens because a simple integration of the gyro outputs does not account for the variation in body-frame attitude with respect to inertial space over that period. The coning error, $\delta\omega_c$, does not oscillate. Therefore, the attitude solution drifts under a constant coning motion. The higher the frequency of the coning motion and the longer the gyro outputs are integrated, the larger the drift will be.

Sculling motion is synchronized angular oscillation about one axis and linear oscillation about an orthogonal axis, as shown in Figure 5.8. This results in an error in the output of an accelerometer triad. If the angular frequency is ω_s and the acceleration amplitude is \mathbf{a}_j , a false acceleration, $\delta\mathbf{a}_s$, is sensed about the axis orthogonal to θ_i and \mathbf{a}_j . From [1],

$$\delta\mathbf{a}_s = \frac{1}{2} \boldsymbol{\theta}_i \wedge \mathbf{a}_j \cos \phi \left(1 - \frac{\sin \omega_s \tau}{\omega_s \tau} \right) \quad (5.86)$$

Similarly, the sculling error, \mathbf{a}_j , does not oscillate so the navigation solution drifts under constant sculling motion. Again, the drift is larger for longer integration times and higher sculling frequencies.

The coning and sculling errors exhibited by an INS are minimized by setting the navigation equations update rate equal to the IMU output rate. However, where the navigation equations are iterated at a lower rate, the effects of coning and sculling are reduced by accounting for the variation in body-frame orientation by summing the gyro and accelerometer outputs using (5.73) and (5.83). Hence, the higher order terms in these equations are sometimes known as coning and sculling corrections. No coning and sculling corrections are applicable where the navigation equations are iterated at the IMU output rate. Where an IMU samples the gyros and accelerometers at a higher rate than it outputs angular rate and specific force, coning and sculling corrections should be applied by its processor prior to output.

Although long periods of in-phase coning and sculling rarely occur in real systems, the navigation solution can still be significantly degraded by the effects of orthogonal vibration modes. Therefore, coning and sculling motion provides a useful test case for inertial navigation equations. The extent to which the navigation equations design must protect against the effects of vibration depends on both the accuracy requirements and the vibration environment. An example of a high-

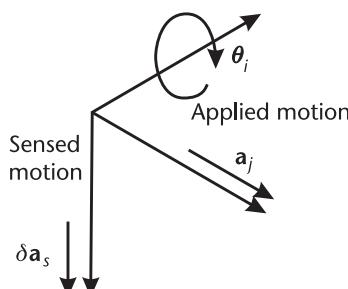


Figure 5.8 Sculling motion. (From: [5]. © 2002 QinetiQ Ltd. Reprinted with permission.)

vibration environment is an aircraft wing pylon, where a guided weapon or sensor pod may be mounted.

5.5 Initialization and Alignment

As Figure 5.1 shows, an INS calculates a navigation solution by integrating the inertial sensor measurements. Thus, each iteration of the navigation equations uses the previous navigation solution as its starting point. Therefore, before an INS can be used to provide a navigation solution, that navigation solution must be initialized. Initial position and velocity must be provided from external information. Attitude may be initialized either from an external source or by sensing gravity and the Earth's rotation.³ The attitude initialization process is also known as alignment because, in a platform INS (see Section 5.7), the inertial instruments are physically aligned with the axes of the local navigation frame.

The initialization is often followed by a period of calibration when stationary or against an external reference, typically lasting a few minutes. This is known as fine alignment, as its main role is to reduce the attitude initialization errors.

5.5.1 Position and Velocity Initialization

The INS position and velocity must be initialized using external information. Where the host vehicle has not moved since the INS was last used, the last known position may be stored and used for initialization. However, an external position reference must be introduced at some point to prevent the navigation solution drift accumulating over successive periods of operation.

INS position may be initialized from another navigation system. This may be another INS, GNSS user equipment, or terrestrial radio navigation user equipment. Alternatively, the INS may be placed near a presurveyed point, or range and/or bearing measurements to known landmarks taken. In either case, the lever arm between the INS and the position reference must be measured. If this is only known in the body frame, the INS attitude will be required to transform the lever arm to the same coordinate frame as the position fix (see Section 2.4.4).

Velocity may be initialized simply by maintaining the INS stationary with respect to the Earth. Alternatively another navigation system, such as GNSS, Doppler radar, or another INS, may be used as a reference. In that case, the lever arm and angular rate are required to calculate the lever arm velocity.

Further problems for velocity initialization are disturbance, vibration, and flexure. For example, when the INS is assumed to be stationary with respect to the Earth, the host vehicle could be disturbed by the wind or by human activity, such as refueling and loading. For ships and boats, water motion is also an issue. For in-motion initialization, the lever arm between the INS and the reference navigation system can be subject to flexure and vibration. The solution is to take initialization measurements over a few seconds and average them. Position can also be affected by flexure and vibration, but the magnitude is usually less than the accuracy required.

5.5.2 Attitude Initialization

Where the INS is stationary, self-alignment can be used to initialize the roll and pitch with all **but the poorest inertial sensors**. However, accurate self-alignment of the **heading** requires **aviation-grade gyros** or better. Heading is often initialized using a magnetic compass, described in Section 10.1.2.

Where the INS is initialized in motion, another navigation system must provide an attitude reference. For guided weapons, the host vehicle's INS is generally used. Multiple-antenna **GNSS** user equipment can also be used to **measure attitude**. However this is very noisy unless **long baselines and/or long averaging times** are used, as described in Section 8.2.2. Another option for some applications is the star tracker, described in Section 11.4.1. In all cases, the accuracy of the attitude initialization depends on how well the relative orientation of the initializing INS and the reference navigation system is known, as well as on the accuracy of the reference attitude. If there is significant flexure in the lever arm between the two systems, such as that which occurs for equipment mounted on an aircraft wing, the relative orientation may only be known to a few tens of milliradians (a degree or two).

For most land vehicles, it can be assumed that the direction of travel defines the body x -axis when the vehicle is not turning (see Section 10.3), enabling a trajectory measured by a positioning system, such as GNSS, to be used to initialize the pitch and heading attitudes. For aircraft and ships, this method will only provide a rough attitude initialization as sideslip, due to wind or sea motion, results in an offset between the heading and the trajectory, while aircraft pitch is defined by the angle of attack needed to obtain lift and ship pitch oscillates due to the sea state.

Self-alignment comprises two processes: a leveling process, which initializes the roll and pitch attitudes, and a gyrocompassing process, which initializes the heading. The leveling is normally performed first.

The principle behind leveling is that, when the INS is stationary (or traveling at constant velocity), the only specific force sensed by the accelerometers is the reaction to gravity, which is approximately in the negative down direction of the local navigation frame at the Earth's surface. Figure 5.9 illustrates this. Thus the attitude, \mathbf{C}_b^n , can be estimated by solving¹

$$\mathbf{f}_{ib}^b = \mathbf{C}_n^b \mathbf{g}_b^n(L_b, h_b) \quad (5.87)$$

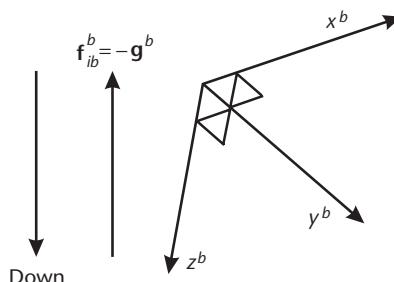


Figure 5.9 Principle of leveling. (From: [5]. © 2002 QinetiQ Ltd. Reprinted with permission.)

given $a_{eb}^\gamma = 0$. Taking the third column of C_b^n , given by (2.14), (5.87) can be expressed in terms of the pitch, θ_{nb} , and roll, ϕ_{nb} , Euler angles:

$$\begin{pmatrix} f_{ib,x}^b \\ f_{ib,y}^b \\ f_{ib,z}^b \end{pmatrix} = \begin{pmatrix} \sin \theta_{nb} \\ -\cos \theta_{nb} \sin \phi_{nb} \\ -\cos \theta_{nb} \cos \phi_{nb} \end{pmatrix} g_{b,D}^n(L_b, h_b) \quad (5.88)$$

where $g_{b,D}^n$ is the down component of the acceleration due to gravity. This solution is overdetermined. Therefore, pitch and roll may be determined without knowledge of gravity, and hence the need for position, using²

$$\theta_{nb} = \arctan \left(\frac{-f_{ib,x}^b}{\sqrt{(f_{ib,y}^b)^2 + (f_{ib,z}^b)^2}} \right) \quad \phi_{nb} = \arctan 2(-f_{ib,y}^b, -f_{ib,z}^b) \quad (5.89)$$

noting that a four-quadrant arctangent function must be used for roll.

Where the INS is absolutely stationary, the attitude initialization accuracy is determined only by the accelerometer errors. For example, a 1-mrad roll and pitch accuracy is obtained from accelerometers accurate to 10^{-3} g. Disturbing motion, such as mechanical vibration, wind effects, and human activity, disrupts the leveling process. However, if the motion averages out over time, its effects on the leveling process may be mitigated simply by time-averaging the accelerometer measurements over a few seconds.

The principle behind *gyrocompassing* is that, when the INS is stationary (or traveling in a straight line in an inertial frame), the only rotation it senses is that of the Earth, which is in the z direction of the ECEF frame. Measuring this rotation in the body frame enables the heading to be determined, except at or very near to the poles, where the rotation axis and gravity vector coincide. Figure 5.10 illustrates the concept. There are two types of gyrocompassing, direct and indirect.

Direct gyrocompassing measures the Earth rotation directly using the gyros. The attitude, C_b^n , may be obtained by solving

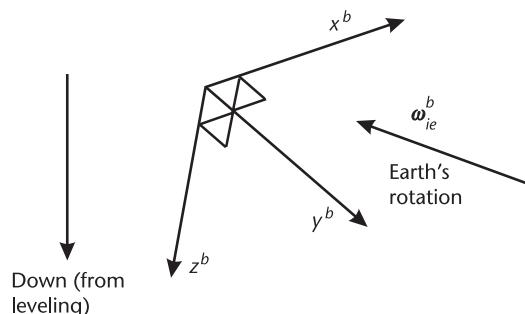


Figure 5.10 Principle of gyrocompassing. (From: [5]. © 2002 QinetiQ Ltd. Reprinted with permission.)

$$\boldsymbol{\omega}_{ib}^b = \mathbf{C}_n^b \mathbf{C}_e^n(L_b, \lambda_b) \begin{pmatrix} 0 \\ 0 \\ \omega_{ie} \end{pmatrix} \quad (5.90)$$

given that $\omega_{eb}^\gamma = 0$. Where the roll and pitch have already been obtained from leveling, the knowledge that the Earth's rotation vector has no east component in the local navigation frame can be used to remove the need for prior position knowledge. Thus, taking the second column of \mathbf{C}_n^b , given by (2.14), substituting this into (5.90) and rearranging gives the heading Euler angle, ψ_{nb} , in terms of the roll, pitch, and gyro measurements:

$$\begin{aligned} \psi_{nb} &= \arctan2(\sin \psi_{nb}, \cos \psi_{nb}) \\ \sin \psi_{nb} &= -\omega_{ib,y}^b \cos \phi_{nb} + \omega_{ib,z}^b \sin \phi_{nb} \\ \cos \psi_{nb} &= \omega_{ib,x}^b \cos \theta_{nb} + \omega_{ib,y}^b \sin \phi_{nb} \sin \theta_{nb} + \omega_{ib,z}^b \cos \phi_{nb} \sin \theta_{nb} \end{aligned} \quad (5.91)$$

Again, a four-quadrant arctangent function must be used. Equations for performing leveling and direct gyrocompassing in one step are presented in a number of texts [1, 9, 10]. However, these require knowledge of the latitude.

In the presence of angular disturbing motion, the gyro measurements used for direct gyrocompassing must be time averaged. However, even small levels of angular vibration will be much larger than the Earth-rotation rate. Therefore, if the INS is mounted on any kind of vehicle, an averaging time of many hours can be required. Thus, the application of direct gyrocompassing is limited.

Indirect gyrocompassing uses the gyros to compute a relative attitude solution, which is used to transform the specific-force measurements into inertial resolving axes. The direction of the Earth's rotation is then obtained from rotation about this axis of the inertially resolved gravity vector. The process typically takes 2 to 10 minutes, depending on the amount of linear vibration and disturbance and the accuracy required. Indirect gyrocompassing is typically combined with fine alignment. A suitable quasi-stationary alignment algorithm is described in Section 13.2.

The accuracy of both gyrocompassing methods depends on gyro performance. Given that $\omega_{ie} \approx 7 \times 10^{-5} \text{ rad s}^{-1}$, to obtain a 1 mrad heading initialization at the equator, the gyros must be accurate to around $7 \times 10^{-8} \text{ rad s}^{-1}$ or about $0.01^\circ \text{ hr}^{-1}$. Only aviation- and marine-grade gyros are this accurate. INS with gyro biases exceeding about $5^\circ/\text{hr}$ are not capable of gyrocompassing at all. Note that the accuracy of the roll and pitch initialization also affects the heading initialization.

The heading initialization error from gyrocompassing [11] is

$$\delta\psi_{nb} = -\frac{\delta f_{ib,y}^b}{g_{b,D}^n} \tan L_b + \frac{\delta\omega_{ib,y}^b}{\omega_{ie}} \sec L_b \quad (5.92)$$

where the accelerometer and gyro error models are presented in Section 4.4.5.

In principle, leveling and gyrocompassing techniques can be performed when the INS is not stationary if the acceleration, \mathbf{a}_{eb}^b , and angular rate, $\boldsymbol{\omega}_{eb}^b$, with respect

to the Earth are provided by an external sensor.³ However, as the relative orientation of the external sensor must be known, this would be no more accurate than simply using the external sensor as an attitude reference.

5.5.3 Fine Alignment

Most inertial navigation applications require attitude to 1 mrad or better, if only to minimize position and velocity drift. Most attitude initialization techniques do not achieve this accuracy. It is therefore necessary to follow initialization with a period of attitude calibration known as fine alignment.¹

In fine alignment techniques, the residual attitude errors are sensed through the growth in the velocity errors. For example, a 1-mrad pitch or roll attitude error will cause the horizontal velocity error to grow at a rate of 10 mm s^{-2} due to false resolving of gravity.

There are three main fine alignment techniques, each providing a different reference to align against. *Quasi-stationary alignment* assumes that the position has been initialized and that the INS is stationary with respect to the Earth and uses zero velocity updates (ZVUs) or integrals thereof. *GNSS alignment*, or INS/GNSS integration, uses position and velocity derived from GNSS and can operate during the navigation phase as well as the alignment phase. Finally, *transfer alignment* uses position or velocity, and sometimes attitude, from another INS or INS/GNSS. It is generally used for aligning a guided-weapon INS between power-up and launch.

In all cases, measurements of the difference between the INS outputs and the reference are input to an estimation algorithm, such as a Kalman filter, which calibrates the velocity, attitude, and sometimes the position, depending on which measurements are used. Inertial instrument errors, such as accelerometer and gyro biases, are often estimated as well. **However, where the INS is stationary, the effects of instrument errors cannot be fully separated from the attitude errors.** For example, a 10 mm s^{-2} accelerometer bias can have the same effect on velocity as a 1-mrad attitude error. To separately observe these errors, **maneuvers must be performed.** For example, if the INS is rotated, a given accelerometer error will have the same effect on velocity as a different attitude error. In quasi-stationary alignment, maneuvers are generally limited to heading changes, with the alignment process suspended during host vehicle maneuvers. For GNSS and transfer alignment, the maneuvers are limited only by the capabilities of the host vehicle. Even with maneuvers, there will still be some correlation between the residual INS errors following fine alignment.

INS/GNSS integration algorithms are described in detail in Chapter 12, while quasi-stationary and transfer alignment are described in Chapter 13. The use of other navigation systems to calibrate INS errors is described in Chapter 14. The main differences between the techniques are the types of measurements used, though all three techniques can use velocity, and the characteristics of the noise on the measurements of differences between the aligning INS and the reference. In quasi-stationary alignment, where zero velocity and angular rate with respect to the Earth are assumed, the main noise source is buffeting of the host vehicle by wind or human activity, such as fuelling or loading. In GNSS alignment, the GNSS receiver

measurements are noisy. In transfer alignment, noise arises from flexure and vibration of the lever arm between the host vehicle's INS and the aligning INS.

Most fine alignment algorithms operate on the basis that position, velocity, and attitude are roughly known at the start of the process. This is important for determining how the system errors vary with time and may allow simplifications, such as the small angle approximation, to be made. For some applications, such as GNSS alignment of tactical-grade INS, there may be no prior knowledge of heading. However, GNSS and transfer alignment algorithms may be adapted to handle this as discussed in Section 12.4.3 [12].

The type of fine alignment technique that is most suitable depends on the application. Where the INS is stationary on the ground, a quasi-stationary alignment is usually best as the noise levels are lowest. Where there is a choice between transfer alignment and GNSS alignment for in-flight applications, the best option is transfer alignment using an INS/GNSS reference, as this combines the higher short-term accuracy and update rate of the INS with the high long-term accuracy of GNSS. Where neither GNSS nor transfer alignment is available, other sensors, such as Doppler radar for aircraft or odometers for land vehicles, may be used (see Chapters 10 and 14).²

5.6 INS Error Propagation

The errors in an inertial navigation system's position, velocity, and attitude solution arise from three sources. These are errors in the accelerometer and gyro measurements, approximations and iteration rate limitations in the navigation equations, and initialization errors. The navigation equations integrate the accelerometer and gyro biases to produce position, velocity, and attitude errors that grow with time. Similarly, the velocity initialization error is integrated to produce a growing position error. Random accelerometer and gyro noise and navigation equations limitations have a cumulative effect on the navigation solution errors. In addition, the attitude errors contribute to the velocity and position errors, and there is both positive and negative feedback of the position errors through the gravity model.

INS error propagation is also affected by the host vehicle trajectory. For example, the effect of scale factor and cross-coupling errors depends on the host vehicle dynamics, as does the coupling of the attitude errors, particularly heading, into velocity and position.

Full determination of INS error propagation is a complex problem and is invariably studied using simulation software. Here, a number of simple examples are presented to illustrate the main principles. These are divided into the short-term and the medium- and long-term cases, followed by a discussion of the effect of circling on error propagation. A more detailed treatment of INS error propagation may be found in a number of inertial navigation texts [1, 7, 11].

Generally, an INS error is simply the difference between an INS-indicated quantity, denoted by a ' \sim ', and the corresponding truth quantity. Thus, the Cartesian position and velocity errors are¹

$$\begin{aligned}\delta \mathbf{r}_{\beta\alpha}^{\gamma} &= \tilde{\mathbf{r}}_{\beta\alpha}^{\gamma} - \mathbf{r}_{\beta\alpha}^{\gamma} \\ \delta \mathbf{v}_{\beta\alpha}^{\gamma} &= \tilde{\mathbf{v}}_{\beta\alpha}^{\gamma} - \mathbf{v}_{\beta\alpha}^{\gamma}\end{aligned}\quad (5.93)$$

while the latitude, longitude, and height errors are

$$\begin{aligned}\delta L_b &= \tilde{L}_b - L_b \\ \delta \lambda_b &= \tilde{\lambda}_b - \lambda_b \\ \delta h_b &= \tilde{h}_b - h_b\end{aligned}\quad (5.94)$$

Coordinate transformation matrices should be used to calculate the attitude error, which is defined by²

$$\delta \mathbf{C}_{\beta}^{\alpha} = \tilde{\mathbf{C}}_{\beta}^{\alpha} \mathbf{C}_{\alpha}^{\beta} \quad (5.95)$$

where the attitude error components are resolved about the axes of the α frame. Note that

$$\begin{aligned}\delta \mathbf{C}_{\alpha}^{\beta} &= \tilde{\mathbf{C}}_{\alpha}^{\beta} \mathbf{C}_{\beta}^{\alpha} = \mathbf{C}_{\alpha}^{\beta} (\delta \mathbf{C}_{\beta}^{\alpha})^T \mathbf{C}_{\beta}^{\alpha} \\ (\delta \mathbf{C}_{\beta}^{\alpha})^T &= \mathbf{C}_{\alpha}^{\beta} \tilde{\mathbf{C}}_{\beta}^{\alpha}\end{aligned}\quad (5.96)$$

where the components of $\delta \mathbf{C}_{\alpha}^{\beta}$ are resolved about the β frame axes.

Except under the small angle approximation, the attitude error in Euler angle form must be computed via coordinate transformation matrices (or quaternions or rotation vectors). Where the small angle approximation applies, the attitude error may be expressed as a vector resolved about an axis of choice. $\delta \boldsymbol{\psi}_{\beta\alpha}^{\gamma}$ is the error in the INS-indicated attitude of frame α with respect to frame β , resolved about the frame γ axes. In terms of the coordinate transformation matrix form of the attitude error,

$$[\delta \boldsymbol{\psi}_{\beta\alpha}^{\alpha} \wedge] \approx \mathbf{I}_3 - \delta \mathbf{C}_{\beta}^{\alpha}, \quad [\delta \boldsymbol{\psi}_{\beta\alpha}^{\beta} \wedge] \approx \delta \mathbf{C}_{\alpha}^{\beta} - \mathbf{I}_3 \quad (5.97)$$

Attitude errors are sometimes known as misalignments. This term is avoided here as it can be confused with the misalignments of the inertial sensor sensitive axes with the body frame that produce cross-coupling errors (see Section 4.4.2).

From Section 4.4.5, the accelerometer and gyro errors are [repeated from (4.17)]:

$$\begin{aligned}\delta \mathbf{f}_{ib}^b &= \tilde{\mathbf{f}}_{ib}^b - \mathbf{f}_{ib}^b \\ \delta \boldsymbol{\omega}_{ib}^b &= \tilde{\boldsymbol{\omega}}_{ib}^b - \boldsymbol{\omega}_{ib}^b\end{aligned}$$

5.6.1 Short-Term Straight-Line Error Propagation

The simplest scenario in which to consider INS errors is short-term propagation when the host vehicle is traveling in a straight line and remains level. In considering

only short-term errors, the effects of curvature and rotation of the Earth and gravity variation may be neglected. Where the host vehicle travels in a straight line, dynamics-induced errors are zero.³

Figure 5.11 shows the position error growth with constant velocity, acceleration, attitude, and angular-rate errors. The position error is simply the integral of the velocity error, so with a constant velocity error,

$$\delta\mathbf{r}_{\beta b}^{\gamma}(t) = \delta\mathbf{v}_{\beta b}^{\gamma} t \quad (5.98)$$

where β is the reference frame and γ the resolving axes. There is no error propagation between axes. As Figure 5.11 illustrates, an 0.1 m s^{-1} initial velocity error produces a 30m position error after 300 seconds (5 minutes).

The velocity error is the integral of the acceleration error, so the following velocity and position errors result from a constant accelerometer bias:

$$\delta\mathbf{v}_{\beta b}^{\gamma}(t) \approx \mathbf{C}_b^{\gamma} \mathbf{b}_a t, \quad \delta\mathbf{r}_{\beta b}^{\gamma}(t) \approx \frac{1}{2} \mathbf{C}_b^{\gamma} \mathbf{b}_a t^2 \quad (5.99)$$

There is no error propagation between axes where the attitude remains constant. As Figure 5.11 shows, an 0.01 m s^{-2} ($\sim 1 \text{ mg}$) accelerometer bias produces a 450m position error after 300s.

Attitude errors, $\delta\psi$, couple into velocity through the specific force. In the short-term straight and level example, the only specific force is the reaction to gravity.

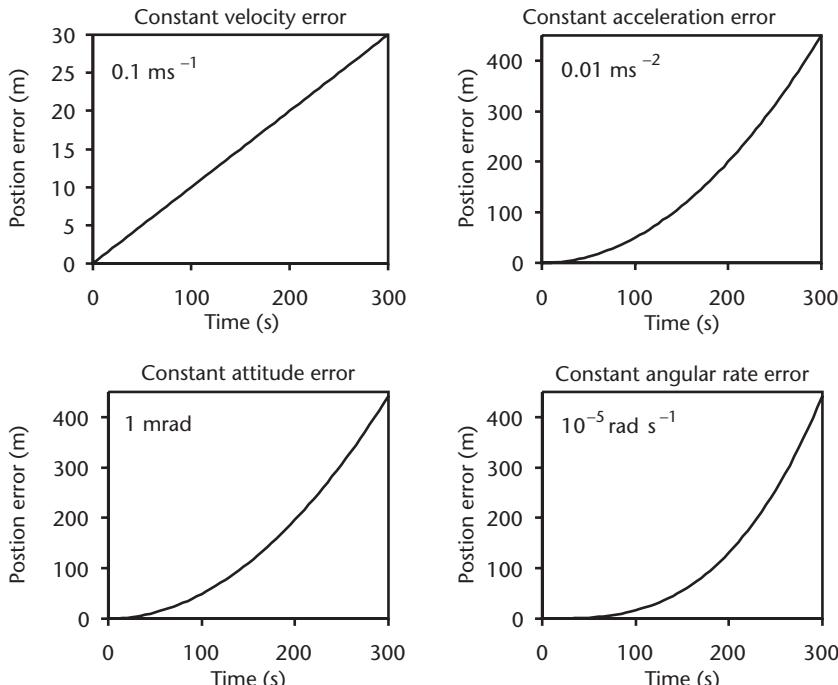


Figure 5.11 Short-term straight-line position error growth.

Thus, pitch (body-frame y -axis) attitude errors couple into along track (body-frame x -axis) velocity errors, and roll (body-frame x -axis) attitude errors couple into across track (body-frame y -axis) velocity errors. The resulting velocity and position errors are thus¹

$$\delta \mathbf{v}_{\beta b}^{\gamma}(t) \approx \delta \boldsymbol{\psi}_{\gamma b}^{\gamma} \wedge \left[\mathbf{C}_b^{\gamma} \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} \right] t = \mathbf{C}_b^{\gamma} \left[\delta \boldsymbol{\psi}_{\gamma b}^b \wedge \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} \right] t \quad (5.100)$$

$$\delta \mathbf{r}_{\beta b}^{\gamma}(t) \approx \frac{1}{2} \delta \boldsymbol{\psi}_{\gamma b}^{\gamma} \wedge \left[\mathbf{C}_b^{\gamma} \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} \right] t^2 = \frac{1}{2} \mathbf{C}_b^{\gamma} \left[\delta \boldsymbol{\psi}_{\gamma b}^b \wedge \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} \right] t^2$$

As Figure 5.11 shows, a 1-mrad (0.057°) initial attitude error produces a position error of ~ 440 m after 300 seconds.

Where the small angle approximation is valid, the attitude error due to a gyro bias, \mathbf{b}_g , is simply

$$\delta \boldsymbol{\psi}_{ib}^b \approx \mathbf{b}_g t \quad (5.101)$$

This leads to velocity and position errors of

$$\delta \mathbf{v}_{\beta b}^{\gamma}(t) \approx \frac{1}{2} \mathbf{C}_b^{\gamma} \left[\mathbf{b}_g \wedge \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} \right] t^2, \quad \delta \mathbf{r}_{\beta b}^{\gamma}(t) \approx \frac{1}{6} \mathbf{C}_b^{\gamma} \left[\mathbf{b}_g \wedge \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} \right] t^3 \quad (5.102)$$

As Figure 5.11 shows, a 10^{-5} rad s $^{-1}$ (2.1° hr $^{-1}$) gyro bias produces a ~ 439 m position error after 300 seconds.²

5.6.2 Medium and Long-Term Error Propagation

The gravity model within the inertial navigation equations, regardless of which coordinate frame they are mechanized in, acts to stabilize horizontal position errors and destabilize vertical channel errors.¹

Consider a vehicle on the Earth's surface with a position error along that surface of δr_b . As a consequence, the gravity model assumes that gravity acts at an angle, $\delta\theta = \delta r / r_{eS}^e$, to its true direction, where r_{eS}^e is the geocentric radius. This is illustrated by Figure 5.12. Therefore, a false acceleration, $\delta \ddot{r}_b$, is sensed in the opposite direction to the position error. Thus the horizontal position error is subject to negative feedback. Assuming the small angle approximation:

$$\delta \ddot{r}_b = -\frac{g}{r_{eS}^e} \delta r_b \quad (5.103)$$

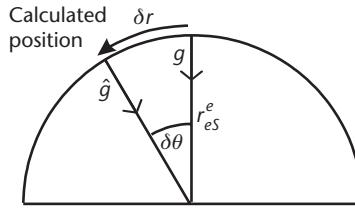


Figure 5.12 Gravity estimation from horizontal position error. (From: [5]. © 2002 QinetiQ Ltd. Reprinted with permission.)

This is the equation for simple harmonic motion with angular frequency $\omega_s = \sqrt{g/r_{eS}^e}$. This is known as the Schuler frequency, and the process as the Schuler oscillation. A pendulum with its pivot at the center of the Earth and its bob at the INS is known as a Schuler pendulum. The period of the Schuler oscillation for a navigation system on the Earth's surface is

$$\tau_s = \frac{2\pi}{\omega_s} = 2\pi \sqrt{\frac{r_{eS}^e}{g}} \quad (5.104)$$

As the strength of the gravity field and the distance from the INS to the center of the Earth varies with height and latitude, this period also varies. At the equator and at the Earth's surface, $\tau_s = 5,974$ seconds (84.6 minutes). Consequently, over periods of order an hour, position errors arising from an initial velocity error, an initial attitude error, or an accelerometer bias are bounded and position errors arising from a gyro bias grow linearly with time, as opposed to cubically. Table 5.2 gives the horizontal position errors arising from different sources for periods of up to about 4 hours [1]. Note that, in practice, instrument biases are not fixed with respect to the north and east axes.²

Figure 5.13 shows the position error magnitude over a 600-second (100-minute) period arising from a 0.1 m s^{-1} initial velocity error, a 0.01 m s^{-2} acceleration error, a 1-mrad initial attitude error, and a $10^{-5} \text{ rad s}^{-1}$ angular rate error. Note that the position error due to the gyro bias is not bounded in the same way as that due to the other error sources. Because of this, much more effort has gone into precision gyro development than precision accelerometer development. So, there

Table 5.2 Medium Term (Up to 4 Hours) Horizontal Position Error Growth from Selected Error Sources³

Error Source	North Position Error, $\delta r_{eb,N}^n$	East Position Error, $\delta r_{eb,E}^n$
Initial velocity error, δv_{eb}^n	$\frac{\sin \omega_s t}{\omega_s} \delta v_{eb,N}^n$	$\frac{\sin \omega_s t}{\omega_s} \delta v_{eb,E}^n$
Fixed accelerometer bias, $(C_b^n b_a)$	$\frac{1 - \cos \omega_s t}{\omega_s^2} (C_b^n b_a)_N$	$\frac{1 - \cos \omega_s t}{\omega_s^2} (C_b^n b_a)_E$
Initial attitude error, $\delta \psi_{nb}^n$	$-(1 - \cos \omega_s t) r_{eS}^e \delta \psi_{nb,E}^n$	$(1 - \cos \omega_s t) r_{eS}^e \delta \psi_{nb,N}^n$
Fixed accelerometer bias, $(C_b^n b_g)$	$-\left(t - \frac{\sin \omega_s t}{\omega_s}\right) r_{eS}^e (C_b^n b_g)_E$	$\left(t - \frac{\sin \omega_s t}{\omega_s}\right) r_{eS}^e (C_b^n b_g)_N$

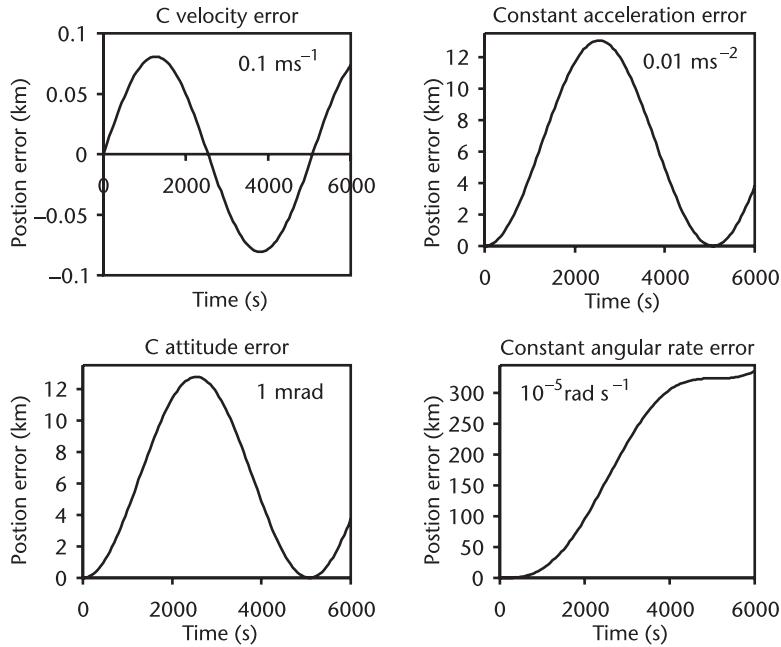


Figure 5.13 Position error growth over a 6,000-second period.

is much greater variation in gyro performance across different grades of INS and IMU.

Where the INS errors are resolved about the local-navigation-frame axes, a further oscillation at the Earth rate, ω_{ie} , and amplitude modulation of the Schuler oscillation at angular frequency $\omega_{ie} \sin L_b$, known as the Foucault frequency, are seen. These are both due to feedback through the Coriolis force terms in the navigation equations. These oscillations are not observed in the ECI-frame INS errors. However, they are present in a local-navigation-frame navigation solution converted from one computed in the ECI frame. Longer-term error propagation is discussed in more detail in [1, 9, 11].

Consider now the vertical channel. As discussed in Section 2.3.5, the gravity varies with height approximately as¹

$$g(h_b) \approx \left(1 - \frac{2h_b}{r_{eS}^e}\right) g_0 \quad (5.105)$$

A positive height error, δh_b , therefore leads to gravity being underestimated. As gravity acts in the opposite direction to that in which height is measured; the virtual acceleration that arises is in the same direction as the height error. Thus,

$$\delta \ddot{h}_b \approx \frac{2g}{r_{eS}^e} \delta h_b \quad (5.106)$$

The vertical position error is subject to positive feedback such that a height initialization error is doubled after about 750 seconds (12.5 minutes). Conse-

quently, for effective long-term vertical navigation, an INS must always be aided by another navigation sensor.²

A barometric altimeter (baro) was always used for vertical aiding prior to the advent of GNSS and still forms a part of many integrated navigation systems. It measures the air pressure and then uses a standard atmospheric model to determine height. It exhibits errors that vary with the weather. A baro's operating principles and error sources are discussed in more detail in Section 10.2.1, while its integration with INS is described in Section 14.3.2.

5.6.3 Errors Due to Circling

Under normal circumstances, the effects of gyro scale factor and cross-coupling errors tend to cancel out as a host vehicle maneuvers. However, where a vehicle starts to circle (e.g., an aircraft surveying an area or waiting in a holding pattern), the effects of these errors start to build up [13]. Typical scale factor and cross-coupling errors for tactical-grade gyros are around 300 ppm. Thus, for each circle completed by the host vehicle, the attitude errors will increase by about 0.1° per axis with tactical-grade gyros, leading to a corresponding growth in position error.¹

An aircraft circling once every 2 minutes with 300-ppm gyro scale factor and cross-coupling errors will build up a ~15-km position error after 30 minutes. With 30-ppm scale factor and cross-coupling errors, typical of RLGs, a ~1.5-km position error is obtained after 30 minutes, still breaking the performance specification for an aviation-grade INS. Consequently, using an INS as the sole means of navigating a circling vehicle cannot be recommended.² A similar problem occurs for guided weapons that spin about their roll axes.

5.7 Platform INS

In a *platform INS*, the accelerometers are mounted on a platform that is connected to the INS casing by 3 or 4 rotatable frames, known as *gimbals*. This is illustrated in Figure 5.14. The gimbals are rotated to maintain alignment of the accelerometer-sensitive axes, with the north, east, and down axes of the local navigation frame or the axes of a wander azimuth frame. For space applications, they may be aligned with the ECI-frame axes. The gyros may be mounted on the platform or at the gimbal axes. This configuration significantly reduces the amount of navigation equations processing that is required.³

As the accelerometer body frame is kept aligned with the coordinate frame used to resolve the navigation solution, the velocity and position may be updated without the need to update the attitude or transform the specific force. The platform configuration also minimizes the effect of instrument errors excited by host vehicle maneuvers.

The gyro outputs, instead of being sent to the navigation equations processor, are used to drive the gimbal rotation to keep the platform aligned with the reference frame as the host vehicle maneuvers. Further gimbal rotation is instigated to account for the Earth rotation and transport rate (see Section 5.3.1). A consequence of this is that many platform INSs do not output angular rate. Also, the attitude of the

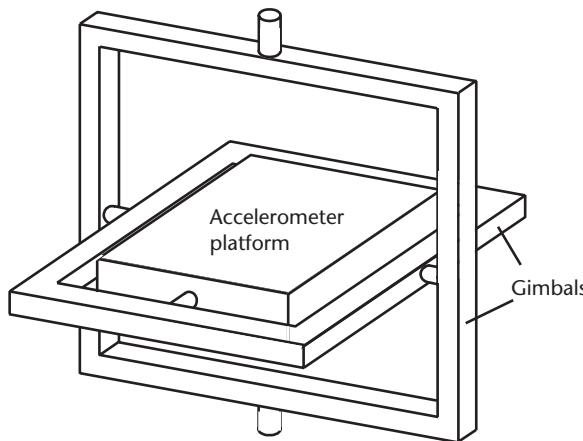


Figure 5.14 Gimbaled platform INS.

INS body with respect to the resolving coordinate frame has traditionally been obtained directly as Euler angles by picking off the gimbal orientations, which has limited accuracy. At initialization, the platform has to be rotated to physically align it with the reference-frame axes.¹

The principal advantage of the platform INS configuration is much reduced processor load, including the option to implement analog computation. Hence, all early INSs were platform systems. Today, processor load is no longer a problem. Strapdown INSs are smaller, mechanically simpler, and lighter than their platform equivalents, so are much cheaper to manufacture. They can also be used to aid airframe stabilization—unlike a platform system, which cannot, as body referenced outputs are required. Consequently, although there are plenty of legacy platform INSs in use aboard aircraft, ships, and submarines, new INSs are all strapdown systems, except for very high precision systems using PIGA accelerometers. Platform INS are discussed in more detail in [14, 15].²

5.8 Horizontal-Plane Inertial Navigation

Road vehicles and trains are constrained to move along the Earth's land surface, while ships and boats are constrained to move along the surface of the sea or a waterway. Thus, navigation for these vehicles is essentially a two-dimensional problem. In principle, inertial navigation could be performed using just three inertial sensors: x - and y -axis accelerometers and a z -axis gyro, saving hardware costs. However, if strapdown sensors are used, they will not remain in the horizontal plane due to terrain slopes or ship pitching and rolling. If the accelerometers are not in the horizontal plane, they will sense the reaction to gravity as well as the horizontal-plane acceleration. A platform tilt of just 10 mrad (0.57°) will produce an acceleration error of 0.1 m s^{-2} . If this is sustained for 100 seconds, the velocity error will be 10 m s^{-1} and the position error 500m. Tilts of 10 times this are quite normal for both cars and boats.

If the inertial sensors are mounted on a freely gimbaled platform, the platform will tilt whenever the vehicle undergoes horizontal acceleration, introducing a large-scale factor error. A stabilized platform is simply a platform INS, requiring six inertial sensors.

In conclusion, inertial navigation should always be performed using a full IMU, regardless of application. However, for many applications, other dead reckoning techniques using lower cost sensors are suitable; these are described in Chapter 10.

References

- [1] Titterton, D. H., and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., Stevenage, U.K.: IEE, 2004.
- [2] Savage, P. G., “Strapdown Inertial Navigation Integration Algorithm Design Part 1: Attitude Algorithms,” *Journal of Guidance Control and Dynamics*, Vol. 21, No. 1, 1998, pp. 19–28; “Strapdown Inertial Navigation Integration Algorithm Design Part 2: Velocity and Position Algorithms,” *Journal of Guidance Control and Dynamics*, Vol. 21, No. 1, 1998, pp. 19–28.
- [3] Wei, M., and K. P. Schwarz, “A Strapdown Inertial Algorithm Using an Earth-Fixed Cartesian Frame,” *Navigation: JION*, Vol. 371, No. 2, 1990, pp. 153–167.
- [4] Farrell, J. L., “Strapdown at the Crossroads,” *Navigation: JION*, Vol. 51, No. 4, 2004, pp. 249–257.
- [5] Groves, P. D., “Principles of Integrated Navigation,” Course Notes, QinetiQ Ltd., 2002.
- [6] Bortz, J. E., “A New Mathematical Formulation for Strapdown Inertial Navigation,” *IEEE Trans. on Aerospace and Electronic Systems*, Vol. AES-7, No. 1, 1971, pp. 61–66.
- [7] Savage, P. G., *Strapdown Analytics Parts 1 and 2*, Maple Plain, MN: Strapdown Associates, 2000.
- [8] Grejner-Brzezinska, D. A., et al., “Enhanced Gravity Compensation for Improved Inertial Navigation Accuracy,” *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 2897–2909.
- [9] Farrell, J. A., and M. Barth, *The Global Positioning System and Inertial Navigation*, New York: McGraw-Hill, 1999.
- [10] Rogers, R. M., *Applied Mathematics in Integrated Navigation Systems*, Reston, VA: AIAA, 2000.
- [11] Britting, K. R., *Inertial Navigation Systems Analysis*, New York: Wiley, 1971.
- [12] Rogers, R. M., “IMU In-Motion Alignment Without Benefit of Attitude Initialization,” *Navigation: JION*, Vol. 44, No. 4, 1997, pp. 301–311.
- [13] Farrell, J. L., “IMU Coast: Not a Silver Bullet,” *Proc. ION 55th AM*, Boston, MA, June 1999, pp. 159–168.
- [14] Tazartes, D. A., M. Kayton, and J. G. Mark, “Inertial Navigation,” in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 313–392.
- [15] Farrell, J. L., *Integrated Aircraft Navigation*, New York: Academic Press, 1976.

Selected Bibliography

- Chatfield, A. B., *Fundamentals of High Accuracy Inertial Navigation*, Reston, VA: AIAA, 1997.
- Jekeli, C., *Inertial Navigation System with Geodetic Applications*, New York: de Gruyter, 2000.
- Wendel, J., *Intergrierte Navigationssysteme: Sensorsdatenfusion, GPS und Inertiale Navigation*, München, Deutschland: Oldenbourg Verlag, 2007.

Endnotes

1. This and subsequent paragraphs are based on material written by the author for QinetiQ, so comprise QinetiQ copyright material.
2. End of QinetiQ copyright material.
3. This paragraph, up to this point, is based on material written by the author for QinetiQ, so comprises QinetiQ copyright material.

Satellite Navigation Systems

Global navigation satellite systems is the collective term for those navigation systems that provide the user with a three-dimensional positioning solution by passive ranging using radio signals transmitted by orbiting satellites. A number of systems aim to provide global coverage. The most well known is the Navigation by Satellite Ranging and Timing (NAVSTAR) Global Positioning System (GPS), owned and operated by the U.S. government and usually known simply as GPS. The Russian GLONASS is also operational. At the time of writing, the European Galileo system was under development, while proposals to provide global coverage for the Chinese Compass system had been announced. In addition, a number of regional satellite navigation systems enhance and complement GNSS.

Some authors use the term GPS to describe satellite navigation in general, while the term GNSS is sometimes reserved for positioning using signals from more than one satellite navigation system. Here, the term GPS is reserved explicitly for the NAVSTAR system, while the term GNSS is used to describe features common to all of the systems. Similarly, the terms GLONASS, Galileo, and so forth are used to describe features specific to those systems.

This chapter provides an introduction to satellite navigation and a description of the individual systems. Section 6.1 describes the basic principles of GNSS, including the system architectures, positioning method, and signal properties. Section 6.2 describes GPS, including its modernization and the six space-based augmentation systems, while Sections 6.3 and 6.4 describe GLONASS and Galileo, respectively. Section 6.5 provides a brief introduction to the regional navigation systems, Beidou, Compass, the Quasi-Zenith Satellite System (QZSS), and the Indian Regional Navigation System (IRNSS). Finally, Section 6.6 compares the different satellite navigation systems and discusses their compatibility and interoperability.

A detailed description of GNSS processing is provided in Chapter 7. This follows the signal path from determination of the satellite positions, through signal propagation and the GNSS receiver, to the generation of the navigation solution and includes discussion of the error sources and the signal geometry. Chapter 8 describes how basic GNSS technology may be enhanced to provide greater accuracy, improved integrity, and better performance in difficult environments.

6.1 Fundamentals of Satellite Navigation

Before the properties of the various satellite navigation systems and services are described, three main concepts must be introduced. First, the architecture of GNSS,

in terms of the space, control, and user segments and their functions, is described. Then the determination of the user position and velocity from ranging measurements is explained and the error sources summarized. Finally, the structure of the GNSS signals and how this is used to obtain ranging measurements is described.

6.1.1 GNSS Architecture

Figure 6.1 shows the architecture of a satellite navigation system, which consists of three components: the space segment, the control or ground segment, and the user segment, which in turn, comprises multiple pieces of user equipment [1–3].

The space segment comprises the satellites, collectively known as a *constellation*, which broadcasts signals to both the control segment and the users. Some authors use the term space vehicle (SV) instead of satellite. GPS, GLONASS, and Galileo satellites are distributed between a number of medium Earth orbits (MEOs), inclined at roughly 60° to the equator with around two orbits per day. Compared to geostationary orbits, these orbits give better signal geometry for positioning (see Section 7.1.4) and better coverage in polar regions.

The ground track of a satellite is the locus of points directly below the satellite on the surface of the Earth. The interval over which it repeats is the lowest common multiple of the Earth rotation period and satellite orbit period.

The signals broadcast by the satellites incorporate both ranging codes and navigation data messages. The ranging codes enable the user equipment to determine the time at which the received signals were transmitted, while a data message includes timing parameters and information about the satellite orbits. A number of atomic clocks aboard each satellite maintain a stable time reference.

The control segment, or ground segment, consists of a network of monitor stations, one or more control stations and a number of uplink stations. The monitor stations obtain ranging measurements from the satellites and send these to the control station(s). The monitor stations are at precisely surveyed locations and

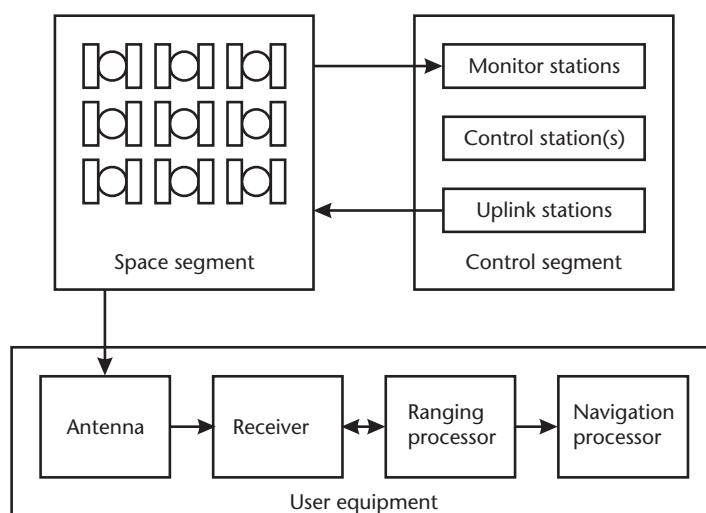


Figure 6.1 GNSS system architecture.

have synchronized clocks, enabling their ranging measurements to be used to determine the satellite orbits and calibrate the satellite clocks. Radar and laser tracking measurements may also be used.

The control stations calculate the navigation data message for each satellite and determine whether any maneuvers must be performed. This information is then transmitted to the space segment by the uplink stations. Most satellite maneuvers are small infrequent corrections, known as station keeping, which are used to maintain their satellites in their correct orbits. However, major relocations are performed in the event of satellite failure, with the failed satellite moved to a different orbit and a new satellite moved to take its place. Satellites are not moved from one orbital plane to another. GPS, GLONASS, and Galileo each maintain an independent control segment. Details of each system's space and control segment are given later in this chapter.

GNSS user equipment is commonly described as GPS, GLONASS, Galileo, and GNSS receivers, as appropriate. However, as Figure 6.1 shows, the receiver forms only part of each set of user equipment. The antenna converts the incoming GNSS radio signals to electrical signals. These are input to the receiver, which demodulates the signals using a clock to provide a time reference. The ranging processor uses acquisition and tracking algorithms to determine the range from the antenna to each of the satellites used from the receiver outputs. It also controls the receiver and decodes the navigation messages. Finally, the navigation processor uses the ranging measurements to compute a position, velocity, and time (PVT) solution. The user equipment is described in detail in Chapter 7.

6.1.2 Positioning

A GNSS position solution is determined by passive ranging in three dimensions [4]. The time of signal arrival, t_{sa} , is determined from the receiver clock, while the time of transmission, t_{st} , of each signal is obtained from its ranging code and data message. Where the receiver and satellite clocks are synchronized, the range, ρ , from a satellite to the user, measured by GNSS user equipment, is obtained by differencing the times of arrival and transmission and then multiplying by the speed of light, c . Thus,

$$\rho_j = (t_{sa,j} - t_{st,j})c \quad (6.1)$$

where the index j is used to denote the satellite number or receiver tracking channel and error sources have been neglected.

Where a ranging measurement from a single satellite is used, the user position can be anywhere on the surface of a sphere of radius ρ centered on that satellite. Where signals from two satellites are used, the locus of the user position is the circle of intersection of two spheres of radii ρ_1 and ρ_2 . Adding a third ranging measurement limits the user position to two points on that circle as illustrated by Figure 6.2. For most applications, only one position solution will be viable in practice; the other may be in space, inside the Earth or simply outside the user's area of operation. Where both solutions are viable, a fourth ranging measurement can be used to resolve the ambiguity.

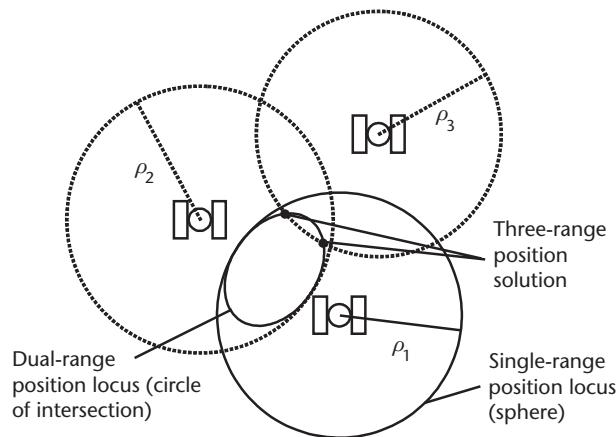


Figure 6.2 Position loci from single, dual, and triple ranging measurements.

In practice, however, the receiver and satellite clocks are not synchronized. If the receiver clock is running ahead of system time, the measured time of arrival, $\tilde{t}_{sa,j}$ will be later than the actual time of arrival, $t_{sa,j}$, resulting in an overestimated range measurement. If the satellite clock is running ahead of system time, the actual time of transmission, $t_{st,j}$, will be earlier than the intended time of transmission, $\tilde{t}_{st,j}$, which is that deduced by the user equipment from the ranging code. This will result in an underestimated range measurement. If the receiver clock is ahead by δt_{rc} and the clock of satellite j ahead by δt_{sj} , the range measurement error, neglecting other error sources, is

$$\begin{aligned}\delta\rho_j &= \tilde{\rho}_{Rj} - \rho_j \\ &= (\tilde{t}_{sa,j} - \tilde{t}_{st,j})c - (t_{sa,j} - t_{st,j})c \\ &= (\tilde{t}_{sa,j} - t_{sa,j})c - (\tilde{t}_{st,j} - t_{st,j})c \\ &= (\delta t_{rc} - \delta t_{sj})c\end{aligned}\quad (6.2)$$

where $\tilde{\rho}_{Rj}$ is the measured range, which is known as the *pseudo-range* to distinguish it from the range in the absence of clock errors. Figure 6.3 illustrates this.

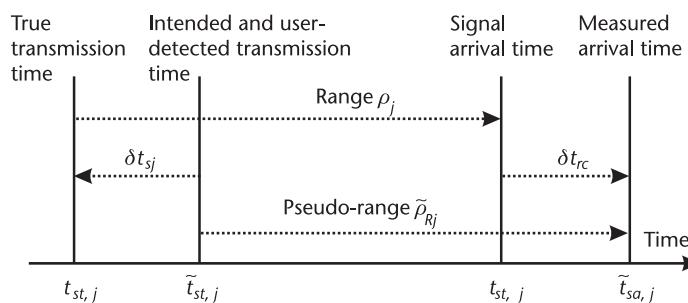


Figure 6.3 Effect of unsynchronized satellite and receiver clocks on range measurement.

The satellite clock errors are measured by the control segment and transmitted in the navigation data message. Therefore, the navigation processor is able to correct for them. The receiver clock offset from system time is unknown. However, as it is common to all simultaneous pseudo-range measurements made using a given receiver, it is treated as a fourth parameter of the position solution to be determined. Therefore, determination of a navigation solution using GNSS requires signals from at least four different satellites to be measured (unless the navigation solution is constrained).

Each pseudo-range measurement, corrected for the satellite clock error (and other known errors), $\tilde{\rho}_{Cj}$, may be expressed in terms of the satellite position, \mathbf{r}_{isj}^i , user antenna position, \mathbf{r}_{ia}^i , and the range error due to the receiver clock error, $\delta\rho_{rc}$, by

$$\tilde{\rho}_{Cj} = \sqrt{(\mathbf{r}_{isj}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa}))^T (\mathbf{r}_{isj}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa}))} + \delta\rho_{rc}(t_{sa}) \quad (6.3)$$

noting that a is used to denote the user-antenna body frame and $\delta\rho_{rc} = \delta t_{rc} c$. The satellite position is obtained from the set of parameters broadcast in the navigation data message describing the satellite orbit, known as the *ephemeris* (see Section 7.1.1), together with the corrected measurement of the time of signal transmission. The four unknowns, comprising the antenna position and receiver clock error, are common to the pseudo-range equations for each of the satellites, assuming a common time of signal arrival. Therefore, they may be obtained by solving simultaneous equations for four pseudo-range measurements. Figure 6.4 illustrates the solution geometry. Similarly, the velocity of the user antenna may be obtained from a set of measurements of pseudo-range rate, the rate of change of the pseudo-range. Calculation of the GNSS navigation solution is described in detail in Section 7.5.

Sources of error in the GNSS navigation solution include differences between the true and broadcast ephemeris and satellite clock errors, signal propagation delays through the ionosphere and troposphere, and receiver measurement errors due to delays in responding to dynamics, receiver noise, radio frequency (RF) interference, and signal multipath. These are all discussed in Section 7.4. The ionosphere and troposphere delays may be partially calibrated using models;

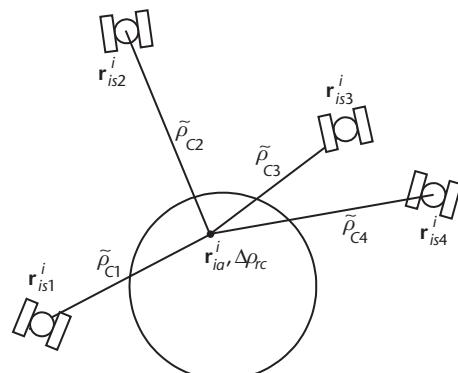


Figure 6.4 Determination of a position solution using four satellite navigation signals.

however, if ranging measurements from a given satellite are made on more than one frequency, the ionosphere propagation delay may be determined from the difference.

GNSS signals may be blocked by buildings, mountainous terrain, and parts of the user equipment's host vehicle. Figure 6.5 illustrates this. Low-elevation satellites are more susceptible to signal blockage, where the elevation is the angle between the horizontal plane and the line of sight from the user to the satellite (see Section 7.1.3). Signal blockage is a particular problem in streets surrounded by tall buildings, known as *urban canyons*. It is the ratio of the height of the buildings to their separation that determines how many GNSS signals get through. Signal blockage is also a problem in mountainous areas, where lower elevation signals will not be receivable in valleys. Consequently, it is not always possible to receive the four signals necessary to compute a position solution, particularly where only one satellite constellation is used. A solution can sometimes be obtained for a limited period with fewer satellites by predicting forward the receiver clock errors or assuming a constant user height.

GNSS position solutions are typically accurate to a few meters, with performance depending on which signals are used, as discussed in Section 7.5.4. Accuracy may be improved by making use of calibration information from one or more reference stations at known locations. This is known as differential GNSS (DGNSS) and is described in Section 8.1. Reference stations may also be used to detect faults in the GNSS signals, a process known as integrity monitoring and discussed in Section 8.5. Integrity monitoring techniques internal to the user equipment are described in Chapter 15.

As well as navigation, GNSS may also be used as a timing service to synchronize a network of clocks. More information is available in [3, 5].

6.1.3 Signals and Range Measurements

GNSS signals are broadcast in a number of frequency bands within the 1–2-GHz L-band region of the electromagnetic spectrum. Multiple signals on different frequencies are used to cater to different user groups, reduce the impact of interference

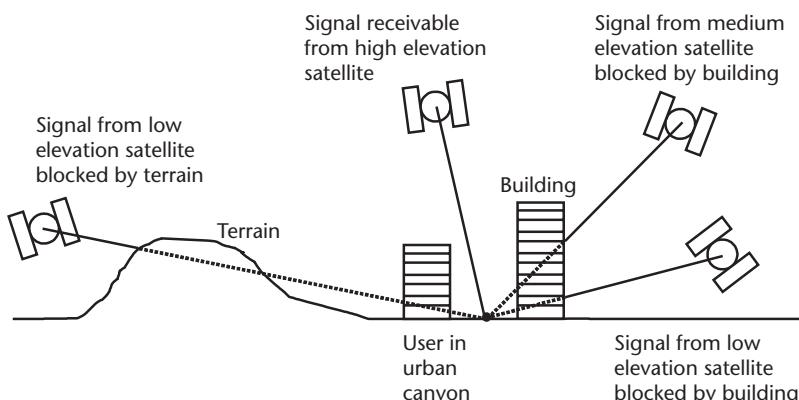


Figure 6.5 Effect of terrain, buildings, and elevation angle.

on one frequency, enable measurement of the ionosphere propagation delay, and aid carrier-phase positioning (see Section 8.2.1). Right-handed circular polarization (RHCP) has been used for all GNSS signals.

The majority of GNSS signals combine a carrier with a navigation data message, D , and a spreading or ranging code, C , using bi-phase shift key (BPSK) modulation. The amplitude of each BPSK signal, s , is given by

$$s(t) = \sqrt{2P}C(t)D(t) \cos(2\pi f_{ca}t + \phi_{ca}) \quad (6.4)$$

where P is the signal power, f_{ca} is the carrier frequency, and ϕ_{ca} is a phase offset [3, 6]. Both C and D take values of ± 1 , varying with time. The data-message rate, f_d , typically varies between 50 and 500 symbol s $^{-1}$, depending on the signal, while the spreading-code rate, f_{co} , varies between 0.511 and 10.23 Mchip s $^{-1}$. Note that it is a matter of convention to describe the data message in terms of symbols and the spreading code in terms of chips; mathematically, the two terms are interchangeable.

The spreading code consists of a pseudo-random noise (PRN) sequence, which is known to the receiver. It is known as a spreading code because multiplying the carrier and navigation data by the code increases the double-sided bandwidth of the signal's main spectral lobe to twice the spreading-code chipping rate while proportionately decreasing the power spectral density. In the receiver, the incoming spread-spectrum signal is multiplied by a replica of the spreading code, a process known as correlation or despreading. If the phase of the receiver-generated spreading code matches that of the incoming signal, the product of the two codes is maximized and the original carrier and navigation data may be recovered. If the two codes are out of phase, their product varies in sign and averages to a low value over time, so the carrier and navigation data are not recovered. Figure 6.6 illustrates this. By adjusting the phase of the receiver-generated PRN code until the correlation peak is found (i.e., the carrier and navigation data are recoverable), the phase of the incoming PRN code is measured. From this, the signal transmission time, $\tilde{t}_{st,j}$, may be deduced. Subtracting this from the signal arrival time, $\tilde{t}_{sa,j}$, provides a measurement of the pseudo-range, \tilde{p}_j . Hence, the PRN code is also known as a ranging code.

The receiver-generated PRN code also spreads interference over the code bandwidth. Following the correlation process, the receiver bandwidth may be reduced to that required to decode the navigation data message, rejecting most of the interference. Consequently, GNSS signals can be broadcast at a substantially lower power per unit bandwidth (after spreading) than thermal noise. Figure 6.7 illustrates the spread-spectrum modulation and demodulation process.

Where the signal and receiver-generated spreading codes are different, the correlation between them is much less than if they are the same and aligned. Consequently, a number of different signals may be broadcast simultaneously on the same carrier frequency, provided they each use a different spreading code. The receiver then selects the spreading code for the desired signal. This is known as code-division multiple access (CDMA) and is in contrast to frequency-division multiple access (FDMA), where each signal has a different carrier frequency, and time-division multiple access (TDMA), where each signal is allocated a different time slot. CDMA is used for GPS, Galileo, Compass, and QZSS signals.

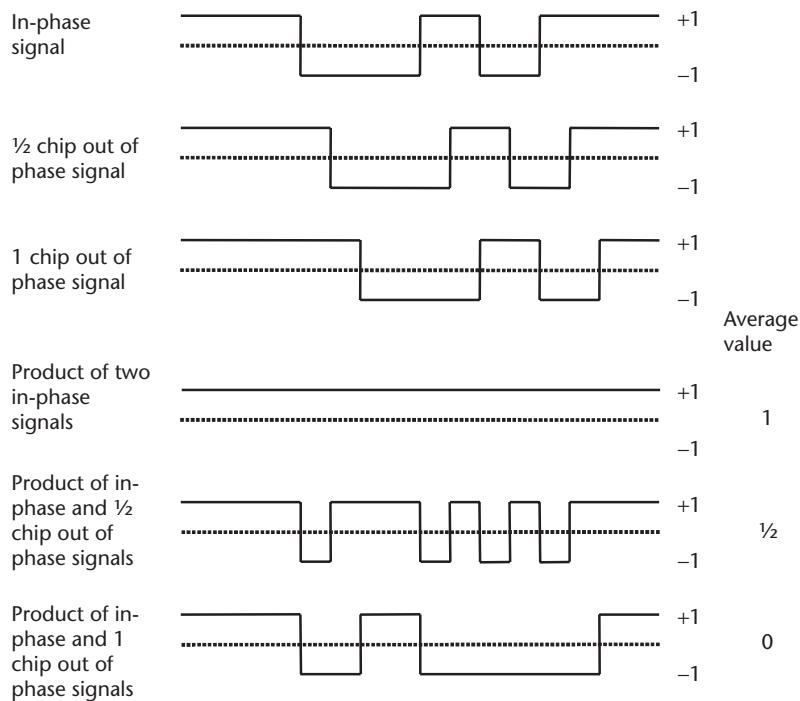


Figure 6.6 Example correlation of pseudo-random noise signals.

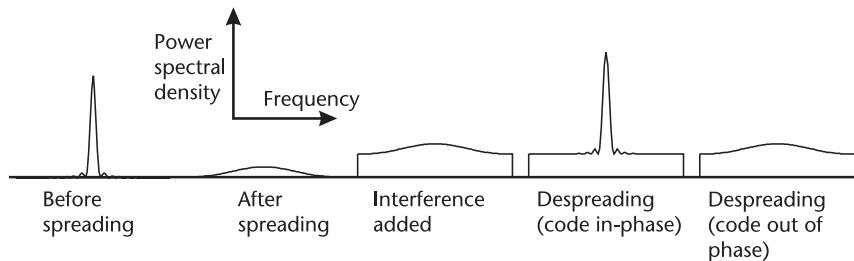


Figure 6.7 Spread-spectrum modulation and demodulation.

Where the pseudo-range is unknown, all phases of the receiver-generated spreading code must be searched until the correlation peak is found, a process known as signal *acquisition*. However, where a pseudo-range prediction from a previous measurement is available, it is only necessary to vary the receiver-generated code phase slightly; this is signal *tracking*. Once the acquisition of a GNSS signal is complete, the user equipment switches to tracking mode for that signal. In most receivers, tracking can operate in much poorer signal-to-noise environments than acquisition. The carrier component of GNSS signals is also tracked and may be used both as an aid to code tracking and to provide low-noise pseudo-range rate measurements. Signal correlation, acquisition, and tracking are described in Sections 7.2.4 and 7.3.

Spreading codes with short repetition lengths, in terms of the number of chips, require fewer options to be searched in the acquisition process. However, there

are false correlation peaks, both between unsynchronized copies of the same code and between the different spreading codes used for cochannel signals in CDMA systems. The shorter the code repetition length, the larger these false peaks are. This conflict may be resolved using layered, or tiered, codes. A short-repetition-length primary code is multiplied by a secondary code, with chip size equal to the repetition length of the primary code, giving a longer repetition length for the full code. The receiver may then correlate the incoming signal with either the primary code or the full code. Figure 6.8 illustrates this.

Higher spreading-code chipping rates offer better resistance against narrowband interference and can sometimes offer higher accuracy positioning (see Section 7.4.3). However, receivers require greater computational capacity to process them.

A faster data message rate enables more information to be broadcast or a given amount of information to be downloaded more quickly. However, faster data rates require a higher postcorrelation bandwidth, reducing interference rejection (see Sections 7.2.4 and 8.3). To resolve this conflict, many GNSS satellites broadcast navigation data on some signals, while leaving others data free [i.e., omitting D in (6.4)]. The data-free signals are sometimes known as pilot signals and can be acquired and tracked with a narrower postcorrelation bandwidth. Data-free signals also give better carrier-tracking performance, as discussed in Section 7.3.3.

Many of the newer GNSS signals use forward error correction (FEC) for the navigation data. This introduces redundancy into the data, enabling correction of decoding errors. Although a higher symbol rate must be used to transmit the message content at a given bit rate, the data message may be successfully decoded in a poorer signal-to-noise environment.

Many of the newer GNSS signals use binary offset carrier (BOC) modulation instead of BPSK. This adds an extra component, the subcarrier, S , giving a total signal amplitude of

$$s(t) = \sqrt{2P}S(t)C(t)D(t) \cos(2\pi f_{ca}t + \phi_{ca}) \quad (6.5)$$

The subcarrier function repeats at a rate f_s , which spreads the signal into two sidebands, centered at $f_{ca} \pm f_s$. To separate the main lobes of these sidebands, f_s

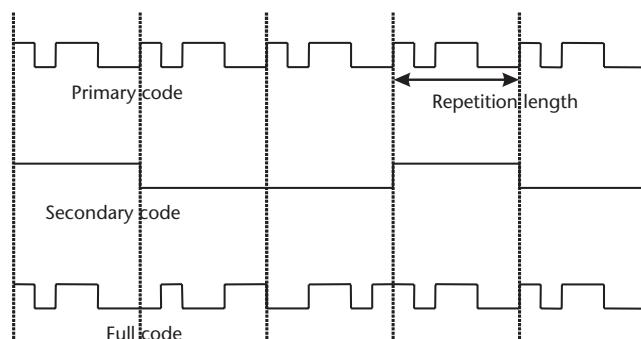


Figure 6.8 A simple layered spreading code.

must be at least the spreading-code chipping rate, f_{co} . BOC modulation can be used to minimize interference with BPSK signals sharing the same carrier frequency. It can also give better code tracking performance than a BPSK signal with the same spreading-code chipping rate [7].

For a basic BOC modulation, the subcarrier function is simply a square wave with chipping rate $2f_s$. This may be sine-phased, in which case the subcarrier function transitions are in-phase with the spreading code transitions, or cosine-phased, where the transitions are a quarter of a subcarrier function period out of phase [8]. More complex subcarrier functions may also be used, such as the alternate BOC used for the Galileo E5 signal (see Section 6.4.2). BOC modulation is described using the shorthand $\text{BOC}_s(f_s, f_{co})$ for sine-phased and $\text{BOC}_c(f_s, f_{co})$ for cosine-phased subcarrier functions, where f_s and f_{co} are usually expressed as multiples of 1.023×10^6 . The terms $\text{BOC}_{\sin}(f_s, f_{co})$ and $\text{BOC}_{\cos}(f_s, f_{co})$ are also used. Figure 6.9 shows the power spectral density of BPSK, sine-phased BOC, and cosine-phased BOC-modulated signals.

Most GNSS signals form part of a multiplex of signals sharing a common carrier frequency. Signal multiplexes are commonly transmitted with in-phase and quadraphase components to maximize transmission efficiency and minimize interference between the signal components. A multiplex of two BPSK signals has a total signal amplitude of

$$s(t) = \sqrt{2P_I} C_I(t) D_I(t) \cos(2\pi f_{ca}t + \phi_{ca}) + \sqrt{2P_Q} C_Q(t) D_Q(t) \sin(2\pi f_{ca}t + \phi_{ca}) \quad (6.6)$$

where the subscript I denotes the in-phase component, and Q denotes the quadraphase component. The combined signal multiplex has quadrature-phase shift key (QPSK) modulation. In the receiver, the in-phase component may be selected by multiplying by $\cos(2\pi f_{ca}t + \phi_{ca})$ and the quadraphase component by multiplying by $\sin(2\pi f_{ca}t + \phi_{ca})$ because the product $\cos(2\pi f_{ca}t + \phi_{ca}) \sin(2\pi f_{ca}t + \phi_{ca})$ averages to zero over a carrier cycle.

6.2 Global Positioning System

NAVSTAR GPS was developed by the U.S. government as a military navigation system. Its controlling body is the GPS Wing, formerly the Joint Program Office

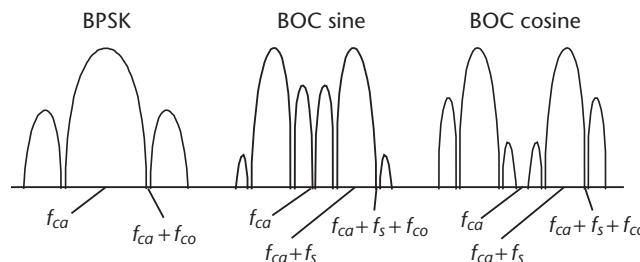


Figure 6.9 Power spectral density of BPSK and BOC-modulated signals (log scale).

(JPO), which operates under the auspices of the Department of Defense (DoD). The JPO was formed in 1973 when a number of military satellite navigation programs, including Transit and the Air Force's 621B program, were merged. The first operational prototype satellite was launched in 1978, IOC was declared at the end of 1993 and full operational capability (FOC) attained at the end of 1994 [9–11].

GPS offers two navigation services, the standard positioning service (SPS), informally known as the civil service, and the precise positioning service (PPS) or military service. The SPS is available to all users with suitable equipment, whereas the PPS is only available to users licensed by the U.S. government, including U.S. and NATO military forces and their suppliers. PPS users have access to encrypted signals that are not available to SPS users.

During the 1990s, the accuracy of the SPS was deliberately degraded to the order of 100m using a technique called selective availability (SA) [12]. This was done in order to deny precise positioning to hostile forces. However, SA can be circumvented using differential GPS (DGPS) (see Section 8.1), while hostile use of GPS can be denied on a local basis using jamming techniques. At the same time, GPS has become an important utility for an increasing number of civil and commercial applications. SA was deactivated on May 1, 2000, but the capability remains.

From the late 1990s, GPS has been undergoing a modernization process. Improvements to the control segment under the Legacy Accuracy Improvement Initiative (L-AII) have increased the accuracy of the ephemeris and satellite clock parameters in the navigation data message, while satellites launched from September 2005 onward have broadcast additional signals to both SPS and PPS users, as described in Section 6.2.2.

Following the L-AII, the basic SPS provides a horizontal accuracy of about 3.8m (1σ) and a vertical accuracy of 6.2m (1σ), while the PPS accuracy is about 1.2m (1σ) horizontally and 1.9m (1σ) vertically (see Section 7.5.4). The modernized SPS will offer a similar accuracy to the PPS.

This section summarizes the GPS space and control segments before describing the signals and navigation data messages. It concludes with a discussion of GPS augmentation systems.

6.2.1 Space and Control Segments

GPS operates with a nominal constellation of 24 satellites and a maximum of 36. All operational satellites provide a full service rather than new satellites being kept on standby until an older satellite fails. There have been a number of generations of GPS satellites, with further generations planned, as listed in Table 6.1. The Block I satellites were prototypes, with the Block II and subsequent satellites providing the operational service. Block IIA and subsequent satellites use momentum management to maintain their orbits with less intervention from the control segment. The Block IIR (“replenishment”) satellites incorporate automatic navigation (Autonav) feature, which enables the navigation data to be updated independently of the control segment using intersatellite ranging measurements. The Block IIR-M (modernized replenishment) and Block IIF (follow-on) satellites broadcast additional signals, as described in Section 6.2.2. The Block III satellites will broadcast at a

Table 6.1 Past, Present, and Future Generations of GPS Satellites

GPS Satellite Block	Launch Dates	Number of Satellites
Block I	1978–1985	10 (Note 1)
Block II	1989–1990	9
Block IIA	1990–1997	19
Block IIR	1997–2004	12 (Note 1)
Block IIR-M	2005–2008	8
Block IIF	From 2008	12–16
Block III	From 2011–2013	24 (planned)

Note 1: Excludes failed launches.

higher power and introduce further new signals. The design life of Block II and IIA GPS satellites is 7.5 years, though most have lasted much longer. The design life of the Block IIR/IIR-M and IIF satellites is 10 years and 12.7 years, respectively. More details can be found in [13].

GPS satellites orbit at a radius of 26,600 km (20,100 km above the Earth's surface) with a period of approximately half a sidereal day (11 hours, 58 minutes) with respect to inertial space, so the ground tracks nearly repeat every sidereal day. The constellation precesses with respect to the Earth's surface by roughly 4 minutes per solar day. The satellites in Blocks II to IIF are arranged in six orbital planes, each inclined at nominally 55° to the equator and separated by 60° of longitude. Figure 6.10 illustrates this. Each plane contains at least four satellites. These are not evenly spaced, with two satellites in each plane separated by about 30° and the others separated by between 92° and 137° where the plane contains the minimum four satellites [14]. This is designed to minimize the effect of a single satellite outage. The constellation configuration for the Block I satellites was different, while that for the Block III satellites has yet to be finalized.

With a 5° masking angle, or minimum elevation, and a typical satellite constellation, including active spares, between 5 and 14 GPS satellites are visible at most times, assuming a clear line of sight. Increasing the masking angle reduces this number. Satellite visibility is higher in equatorial and polar regions than at mid latitudes [14].

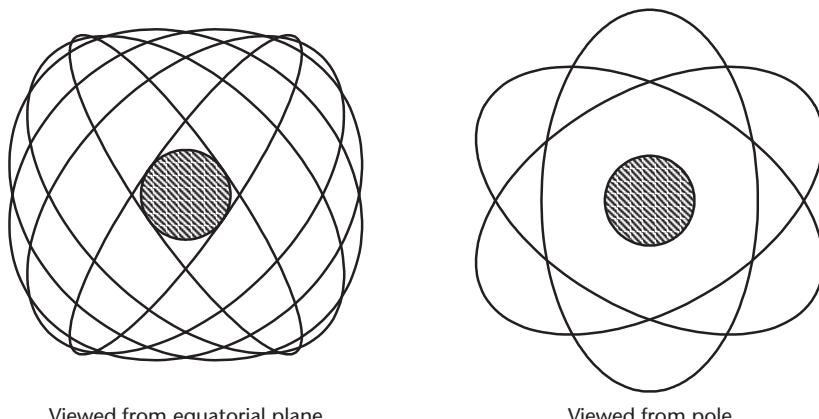


Figure 6.10 GPS satellite orbits. (From: [15]. © 2002 QinetiQ Ltd. Reprinted with permission.)

The original GPS operational control segment (OCS) comprised a master control station (MCS) near Colorado Springs, Colorado, five (later six) monitor stations, and four uplink stations [16]. This is being upgraded under the L-AII. By incorporating NGA stations, the monitoring network was extended to 12 stations in 2005 [17], with a further extension to 17 underway at the time of writing. In addition, the MCS software is being upgraded and an alternate MCS added in California.

Further control segment enhancements are planned for introduction with the launch of the Block III satellites, including implementation of a full integrity monitoring and alert system, covering all of the signals transmitted. This phase of the GPS modernization program is collectively known as GPS III.

6.2.2 Signals

There are 10 different GPS navigation signals, broadcast across three bands, known as link 1 (L1), link 2 (L2), and link 5 (L5). The carrier frequencies are 1575.42 MHz for L1, 1227.60 MHz for L2, and 1176.45 MHz for L5, while the declared double-sided signal bandwidth is 30.69 MHz in each band. The signals are summarized in Table 6.2 and their PSDs illustrated by Figure 6.11. However, many of these signals are being introduced under the GPS modernization program and are not broadcast by all satellites while the L1C signals will not be broadcast before 2011. GPS satellites can also transmit a signal on 1381.05 MHz (L3), but this is used by the Nuclear Detonation Detection System; it is not designed for navigation [6].

The nominal signal powers listed in Table 6.2 are minimum values. Satellites initially transmit at higher powers, but the power drops as the satellite ages. The Block III satellites will transmit at higher power with equal power in the L1 and L2 bands.

Table 6.2 GPS Signal Properties

Signal	Band and Carrier Frequency (MHz)	Service	Modulation and Chipping Rate ($\times 1.023 \text{ Mchip s}^{-1}$)	Navigation Message Rate (symbol s^{-1})	Minimum Received Signal Power (dBW)	Satellite Blocks
C/A	L1, 1575.42	SPS/PPS	BPSK 1	50	-158.5	All
P(Y)	L1, 1575.42	PPS	BPSK 10	50	-161.5	All
M code	L1, 1575.42	PPS	BOC _s (10,5)	Note 2	Note 2	From IIR-M
L1C-d	L1, 1575.42	PPS	BOC _s (1,1)	Note 1	100	-163
L1C-p	L1, 1575.42	PPS	BOC _s (1,1)	Note 1	None	-158.3
L2C	L2, 1227.60	SPS	BPSK 1	50	-160	From IIR-M
P(Y)	L2, 1227.60	PPS	BPSK 10	50	-164.5	All
M code	L2, 1227.60	PPS	BOC _s (10,5)	Note 2	Note 2	From IIR-M
L5I	L5, 1176.45	SPS	BPSK 10	100	-158	From IIF
L5Q	L5, 1176.45	SPS	BPSK 10	None	-158	From IIF

Note 1: This provisional L1C modulation scheme was recently replaced by a modified binary offset carrier (MBOC), comprising a 10/11 power BOC_s(1,1) modulation and 1/11 power BOC_s(6,1) modulation [18].

Note 2: Some details of the M-code signal are not in the public domain. PPS users are directed to the relevant interface control document (ICD) [19].

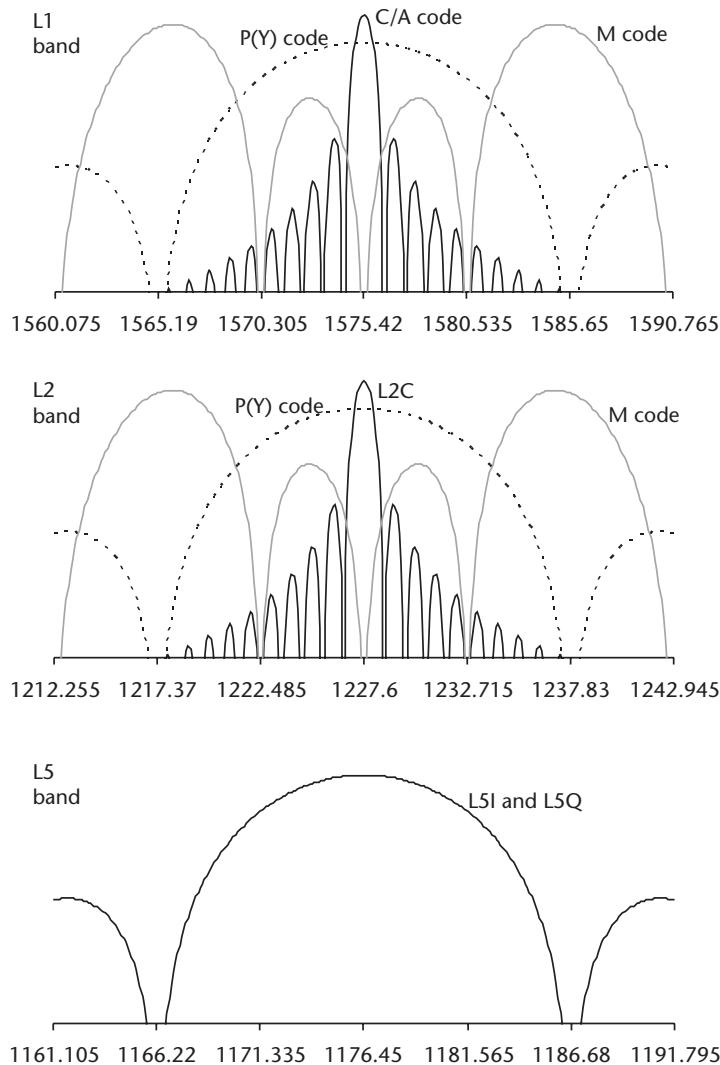


Figure 6.11 GPS signal multiplex power spectral densities (log scale).

The coarse/acquisition (C/A) and precise (encrypted precise) (P(Y))-code signals are known as the legacy GPS signals, as they predate the modernization program. The C/A code is so named because it was intended to provide less accurate positioning than the P(Y) code and most PPS user equipment acquires the C/A-code signal before the P(Y)-code signals (see Section 7.3.1). The C/A code is 1,023 chips long and repeats every millisecond (i.e., at 1 kHz). This makes it relatively easy to acquire; however, the correlation properties are poor, with cross-correlation peaks only 21–24 dB below the main autocorrelation peak. Each satellite transmits 1 of 36 C/A code sequences (originally 31), each allocated a PRN signal number and selected from a family of Gold codes to minimize cross-correlation [6]. Each Gold code is $2^n - 1$ chips long, where n is an integer. The $1.023 \text{ Mchip s}^{-1}$ chipping rate for the C/A code was thus chosen to give a 1-ms repetition interval.

The encrypted precise (Y) code comprises the publicly known precise (P) code multiplied by an encryption code, which is only available to licensed PPS users. This encryption acts as an antspoofing (AS) measure because it makes it difficult for hostile forces to deceive GPS user equipment by broadcasting replica signals, which is known as spoofing. All GPS satellites normally broadcast Y code, but can be switched to broadcast P code. P code is also used by GPS signal simulators to test PPS user equipment without the need for encryption data. The notation P(Y) code is commonly used to refer to the P and Y codes collectively. The P code is a product of two PRN codes, one of length 15,345,000 chips (1.5 seconds) and the other 37 chips longer, giving a total code period of just over 38 weeks. Each GPS satellite transmits a different 1-week repeating segment of this code on both the L1 and L2 frequencies [6].

Broadcasting SPS signals on more than one frequency improves accuracy and robustness. The link 2 civil (L2C) signal is a time-division multiplex (TDM) of two components, one carrying the civil moderate (CM) code with a navigation data message and the other carrying the civil long (CL) code data free. The chipping rate of each code is $511.5 \text{ kchip s}^{-1}$. The L2C signal comprises alternate chips from each component, giving a total rate of $1.023 \text{ Mchip s}^{-1}$. The CM code is 10,230 chips long and has a repetition period of 20 ms, matching the navigation message symbol length. The CL code is 75 times longer at 767,250 chips and repeats every 1.5 seconds [20, 21]. A receiver can use either the CM code, the CL code, or both. The CM code can be acquired more quickly or with less processing power than the CL code, while the data-free CL code gives more accurate carrier tracking and better performance in poor signal-to-noise environments. The CM code is more difficult to acquire than the C/A code but offers an increased margin of 45 dB between the main autocorrelation peak and cross-correlation peaks. The L2C signal is not suited to safety-of-life applications because of the amount of interference in the L2 band.

The new military (M)-code signals are the first GNSS signals to use BOC modulation. This is done to provide spectral separation from the SPS signals, enabling use of jamming to prevent hostile use of GPS and allowing higher power PPS signals to be broadcast without disrupting the civil GPS service [22]. The M code is intended to be acquired directly, despite the use of a long code length and high chipping rate. The processing load can be reduced by single sideband acquisition (see Section 7.3.1). A number of acquisition aids have also been investigated.

The original M-code design has been modified to incorporate time-division data multiplexing (TDDM), whereby the navigation data message is only modulated on alternate chips of the PRN code, enabling the user to obtain the benefits of a data-free signal component. As with L2C, the receiver may use either signal component or both [23].

The Block III satellites may have the capability to broadcast a *spot beam* of additional M code signals over a limited area, providing a received signal strength 20 dB higher than the global M-code signal. The spot beam would broadcast in both the L1 and L2 bands using different PRN codes.

The L5 signal multiplex comprises two $10.23 \text{ Mchip s}^{-1}$ SPS signals. The in-phase signal, L5I, incorporates a navigation data message, while the quadrature

signal, L5Q, is data-free. Both signals use layered ranging codes. The primary codes are 10,230 chips long with a 1-ms repetition period. Different codes are used on the in-phase and quadrature signals. These are multiplied by 1 kchip s^{-1} Neuman-Hoffman codes to give full codes known as SpV codes. The L5I SpV codes repeat every 10 ms, the navigation-message symbol length, while the L5Q codes repeat every 20 ms [21, 24].

The L1C data-modulated signal will use a 10,230 chip code, repeating every 10 ms. The data-free L1C signal will be layered, with the same 10,230-chip primary code and a 1,800-chip secondary, or overlay, code, giving a total repetition period of 18 seconds [25].

6.2.3 Navigation Data Messages

GPS satellites broadcast three different navigation data messages. The legacy navigation message is broadcast simultaneously on the C/A and both P(Y)-code signals, MNAV messages are broadcast on the M-code signals, and CNAV messages are broadcast on the L2C signal (CM component) and L5I signal. A further, C2NAV, message will be introduced with the L1C signals.

The legacy navigation message is broadcast in a fixed-frame format with no FEC at a data rate of 50 bit s^{-1} . It is divided into 30-bit words of 0.6-second duration, each incorporating a parity check, while the full message lasts 12.5 minutes. A full description of the message can be found in the interface standard (IS) [26, 27]; here, a brief summary is presented.

The satellite clock calibration data (see Section 7.4.1) and ephemeris information, expressed as a set of 16 Keplerian orbital parameters (see Section 7.1.1), for the transmitting satellite are broadcast every 30 seconds. Issue of data ephemeris (IODE) and issue of data clock (IODC) integers are incremented each time this navigation data is updated, currently every 2 hours. The handover word (HOW), which aids the transition from C/A code to P(Y) code tracking by indicating the number of 1.5-second P(Y)-code periods that have occurred thus far in the week, is transmitted every 6 seconds.

The almanac data is only broadcast every 12.5 minutes. It comprises the approximate ephemeris parameters, clock calibration, and health of the entire satellite constellation, up to a maximum of 32 satellites. It is intended to aid the user equipment in selecting which satellites to use and acquiring the signals. The almanac data is valid for longer periods than the precise ephemeris data, giving satellite positions to an accuracy of 900m up to a day from transmission, 1,200m for one week, and 3,600m for up to two weeks. The health information comprises an 8-bit index per satellite indicating the status of the signals and navigation data. Also broadcast every 12.5 minutes are the eight coefficients of the ionosphere propagation delay correction model for single-frequency users (see Section 7.4.2) and GPS-UTC time conversion data. GPS time is synchronized with Universal Coordinated Time (UTC) as maintained by the U.S. Naval Observatory (USNO). However, it is not subject to leap seconds and is expressed in terms of a week number and the number of seconds from the start of that week (midnight Saturday/Sunday). The week number “rolls over” every 1,024 weeks (19 years and 227/228 days).

The MNAV and CNAV messages are structured differently from the legacy navigation message. Their subframes, which are headed by a message type indicator, may be transmitted in any order. This provides greater flexibility to transmit different information at different rates and enables changes in satellite health to be alerted to users much more quickly. MNAV subframes are 400 bits long, and CNAV subframes are 300 bits long. The CNAV message uses 1/2-rate FEC, giving a data rate of half the symbol rate; thus, the data rate is 25 bit s^{-1} on L2C and 50 bit s^{-1} on L5I. The MNAV message broadcasts higher precision ephemeris and satellite clock parameters than the legacy message, and there are plans to improve the precision of these parameters in the CNAV message as well. A full description of the messages can be found in the relevant ICD/IS [19, 26].

6.2.4 Augmentation Systems

Augmentation systems supplement GPS with additional ranging signals, a differential corrections service, and integrity alerts. The additional ranging signals increase coverage in urban and mountainous areas, where many GPS signals may be blocked by terrain or buildings. They also assist user-equipment-based integrity monitoring (see Chapter 15). Differential corrections enable more accurate positioning using DGPS (see Section 8.1), while the integrity alerts (see Section 8.5) protect users from the effects of erroneous GPS signals, which is essential for safety-critical applications, such as civil aviation. At the time of writing, the GPS OCS only monitored the health of the PPS signals, and it could take over 2 hours to alert users of problems due to limitations of the uplink schedule and legacy navigation message format. These issues are being addressed as part of the modernization program.

There are two main types of augmentation systems. Space-based augmentation systems are designed to serve a large country or small continent and broadcast to their users via geostationary satellites. Ground-based augmentation systems (GBAS) serve a local area, such as an airfield, providing a higher precision service than SBAS and broadcasting to users via ground-based transmitters.

There are six SBAS systems, at varying stages of development at the time of writing, as summarized in Table 6.3 [28, 29]. Each uses a network of several tens of reference stations across its coverage area to monitor the GPS signals. The full SBAS service, including differential corrections and ionosphere data, is only available in the region spanned by the reference stations. However, the additional ranging signals and satellite signal failure alerts can be used throughout the coverage area of each geostationary satellite, which typically spans latitudes from -70° to $+70^\circ$ and longitudes within 70° of the satellite. The full-service coverage area of an SBAS system may be expanded within the signal footprint by adding additional reference stations. WAAS coverage is being expanded from just the United States to incorporate Mexico and most of Canada, while EGNOS coverage may be extended into Russia.

All SBAS systems broadcast a common signal format, originally developed for WAAS, enabling the same receivers to be used. A signal is broadcast on the L1 carrier frequency with the same chipping rate and code length as C/A code, but different PRN codes and a different navigation data message. As part of a program

Table 6.3 Satellite-Based Augmentation Systems

SBAS	<i>Full Service Coverage Area</i>	<i>Geostationary Satellite Longitude(s)</i>	IOC
WAAS	North America	-178°, -142°, -133°, -107°	2003 (US)
EGNOS	Europe and surrounding countries	-15.5°, 21.5°, 25°	2006
Multifunction transport satellite (MTSat) augmentation system (MSAS)	Japan	140°, 145°	2007
GPS/GLONASS and GEO augmented navigation (GAGAN)	India	34°, 83°, 132°	
Satellite navigation augmentation system (SNAS)	China	80°, 110°, 140°	
Nigerian communications satellite (NIGCOMSAT)	Nigeria	42°	

of enhancements to the SBAS standard to meet the WAAS FOC requirements, a second signal, based on the GPS L5I signal, is being added.

The SBAS navigation message on the L1 frequency is broadcast at 500 symbol s⁻¹ with 1/2-rate FEC, giving a data rate of 250 bit s⁻¹. There are a number of different message types, each 250 bits long and taking 1 second to transmit. These can be transmitted in any order and are detailed in the ICD [30]. The data includes SBAS satellite position and velocity, differential corrections for the GPS signals, ionosphere model parameters, and data that can be used to estimate the accuracy of the SBAS-corrected pseudo-range measurements. Fast corrections messages, normally transmitted every 10 seconds, allow the differential corrections, accuracy, and satellite health data to be updated rapidly. In the event of a rapid satellite signal failure, the fast corrections message can be brought forward to provide an integrity alert to the user within 6 seconds of detection. More details on SBAS can be found in [31].

GBAS, sometimes known as the integrity beacon landing system (IBLS), has yet to be deployed at the time of writing, though a lot of research and development has been undertaken under the local area augmentation system (LAAS) program in the United States [32]. The differential corrections and integrity data are broadcast on a VHF channel, while additional ranging signals are provided by ground-based GPS-like transmitters, known as pseudolites.

A major issue with the use of pseudolites with a C/A code-like signal is that a received pseudolite signal more than about 20 dB stronger than the GPS C/A-code signal can prevent reception of the GPS signal, while the reverse is the case where the pseudolite signal is more than 20 dB weaker than the GPS signal. This is known as the *near-far problem* and limits the effective range of C/A code pseudolites. The problem may be mitigated by pulsing the pseudolite signal [33].

Australia is developing a hybrid augmentation system, known as the ground-based regional augmentation system (GRAS) [34]. It combines an SBAS-like reference station network with a network of ground-based VHF transmitters, mostly

located near airfields, each broadcasting to airborne users within a radius of about 350 km.

In civil aviation, integration of GNSS with inertial navigation (Chapter 12), other sensors, such as a barometric altimeter (Chapter 14), and aircraft-based integrity monitoring (Chapter 15) are collectively known as an aircraft-based augmentation system (ABAS).

6.3 GLONASS

GLONASS, Global'naya Navigatsionnaya Sputnikovaya Sistema, was developed as a military navigation system by the USSR from the mid 1970s, in parallel to GPS. Like GPS, it was designed to offer both a civil and a military positioning service. The first satellite was launched in 1982. Following the dissolution of the Soviet Union, GLONASS development was continued by Russia, with a full satellite constellation achieved in 1995. However, due to financial problems and the relatively short lifetime of the satellites, the constellation was then allowed to decay, reaching a nadir of seven satellites in 2001.

In August 2001, a modernization program was instigated, rebuilding the constellation, introducing new signals, and updating the control segment. IOC with 18 satellites was scheduled for the end of 2007, while a full constellation is planned for the end of 2009. Since December 2004, India has been a partner in the operation of GLONASS [35].

As for GPS, this section describes the space and control segments, followed by the signals and navigation data messages. GAGAN will provide augmentation for GLONASS as well as GPS, while plans to add GLONASS data to the EGNOS transmissions were also under consideration at the time of writing.

6.3.1 Space and Control Segments

The full GLONASS constellation comprises 24 satellites, of which 3 are active spares. There are three generations of GLONASS satellites. More than 80 of the original GLONASS satellites have been launched between 1982 and 2007. These satellites have a design life of 3 years, but have lasted for an average of 4.5 years in practice. The first of the modernized GLONASS-M satellites was launched in 2003, featuring an additional civil signal, higher accuracy clock, and an extended lifetime of 7 years. GLONASS-M is an intermediate design, and only 11 satellites are planned. The fully modernized GLONASS-K satellites will have a lifetime of at least 10 years and will broadcast further new signals. The first GLONASS-K launch is scheduled for 2009, with a total of 27 satellites planned [35].

GLONASS satellites orbit at a radius of 25,600 km (19,100 km above the Earth's surface) with a period of 11 hours, 15 minutes, so each satellite completes $2\frac{1}{2}$ orbits per sidereal day. There are 24 satellite slots, uniformly spaced in 3 orbital planes, 120° apart in longitude with an inclination of 64.8° to the equator. Thus, the ground track of each individual satellite repeats every 8 sidereal days, but each ground track is performed by one of the satellites in the constellation each day. Figure 6.12 shows the satellite orbits.

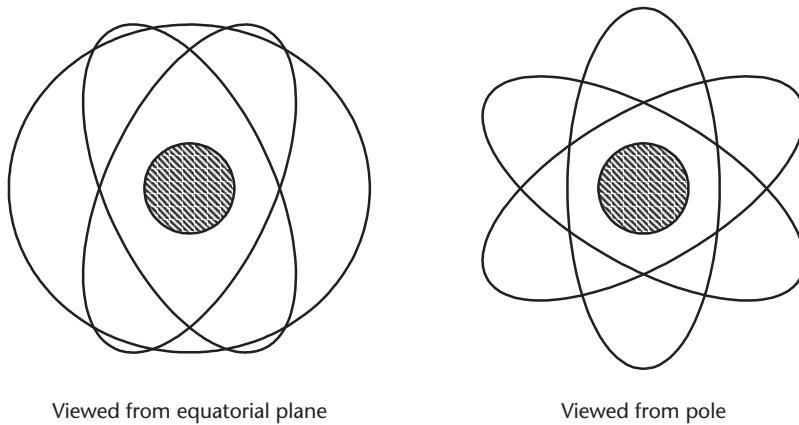


Figure 6.12 GLONASS satellite orbits.

The GLONASS ground-based control complex (GBCC) is largely limited to Russian territory. It comprises a system control center near Moscow, two monitor stations in Moscow, and four telemetry, tracking and control (TT&C) stations. The TT&C stations uplink commands to the satellites and also perform radar tracking every 10 to 14 orbits to measure the satellite orbits, while the monitor station measurements are used only to calibrate the satellite clocks. Laser tracking stations are used to periodically calibrate the radar tracking [36].

As part of the modernization program, GLONASS will change to determining the satellite orbits from the GLONASS ranging signals using a network of at least 12 monitor stations [34]. The space and control segment modernization should enable GLONASS to achieve a similar positioning accuracy to GPS by around 2010.

6.3.2 Signals

The GLONASS signals are summarized in Table 6.4. As with GPS, some signals are only broadcast by the newer satellites. The P code was originally intended for military users only and can be encrypted, though this has not been done in practice.

Table 6.4 GLONASS Signal Properties

Signal	Band and Frequency Range (MHz)	Modulation and Chipping Rate ($Mchip\ s^{-1}$)	Navigation Message Rate ($symbol\ s^{-1}$)	Minimum Received Signal Power (dBW)	Satellite Blocks
C/A	L1, 1592.95–1613.86	BPSK 0.511	50	-161	All
P code	L1, 1592.95–1613.86	BPSK 5.11	50	-161	All
C/A	L2, 1237.83–1256.36	BPSK 0.511	50	-167	From GLONASS-M
P code	L2, 1237.83–1256.36	BPSK 5.11	50	-167	All
Note 1	L3, 1190.5–1212	BPSK 4.095	Note 1	Note 1	From GLONASS-K
Note 1	L3, 1190.5–1212	BPSK 4.095	None	Note 1	From GLONASS-K

Note 1: Full details of the L3-band signals were unknown at the time of this writing.

The GLONASS L1 and L2 bands lie just above their GPS counterparts, while the L3 band lies between the GPS L5 and L2 bands [35].

The C/A code has a 511-chip length, 1-ms repetition period, and 511-kchip s^{-1} chipping rate, while the P code has a 5.11-Mchip s^{-1} chipping rate and is truncated to give a 1s repetition period. Unlike, GPS, GLONASS uses frequency-division multiple access, with each satellite broadcasting the same PRN codes on different carrier frequencies. Each satellite is allocated a channel number, k , and broadcasts on $1602 + 0.5625k$ MHz in the L1 band and $1246 + 0.4375k$ MHz in the L2 band. Until September 1993, each satellite was allocated its own channel. After that, only channels 1 to 12 were used to limit interference to radio astronomy, with satellites in opposite slots in the same orbital plane sharing the same channels. This only causes interference to space-based users. A further migration to channels -7 to $+6$ has been proposed to further reduce interference to radio astronomy. Use of FDMA significantly reduces the cross-correlation between signals using short ranging codes, but requires more complex receivers [36].

In the L3 subband, in-phase and quadrature BPSK signals will be broadcast at $4.095\text{ Mchip s}^{-1}$. Channel allocations of $1201.5 + 0.421875k$ are currently proposed [35]. However, use of CDMA for future GLONASS signals is also under consideration [37]. Adding a further new signal based on GPS L1C is also an option.

6.3.3 Navigation Data Messages

GLONASS broadcasts different navigation data messages on the C/A code and P code signals. Both messages employ a fixed-frame format with no FEC and a data rate of 50 bit s^{-1} . The messages are divided into lines of 100 bits, lasting 2 seconds, each with a parity check. The full C/A-code message repeats every 2.5 minutes, while the P-code message repeats every 12 minutes. The ephemeris and satellite clock information for the transmitting satellite is broadcast every 30 seconds for C/A code and 10 seconds for P code, while the almanac is repeated at the full message rate. GLONASS does not broadcast ionosphere model parameters. The ephemeris for the transmitting satellite is expressed simply as an ECEF-frame position, velocity, and acceleration with a reference time, rather than as Keplerian parameters. The user equipment determines the current position and velocity by simple interpolation. The broadcast ephemeris parameters are updated every 30 minutes [38].

A new navigation data message will broadcast in the L3 band. This may include integrity information and differential corrections [35].

6.4 Galileo

Development of the Galileo satellite navigation system was initiated in 1999 by the European Union (EU) and European Space Agency (ESA). The first test satellite, GIOVE-A, was launched on December 28, 2005, to be followed by further test satellites, GIOVE-B and possibly GIOVE-A2, in 2007–2008. The first four satellites of the permanent constellation for the in-orbit validation (IOV) phase of develop-

ment will launch next, followed by full constellation deployment. Initial operational capability is planned for 2010–2012 with full operational capability by 2014.

Unlike GPS and GLONASS, Galileo has been developed as a purely civil navigation system. It is managed by the GNSS Supervisory Authority (GSA). Development is funded mainly by the EU, while user charges are proposed for some of Galileo's services. A number of non-European countries are also participating, but will not be involved in critical aspects of the program.

Galileo offers four navigation services: an open service (OS), a safety-of-life (SOL) service, commercial services (CS), and a public regulated service (PRS) [39, 40]. In addition, a search-and-rescue (SAR) service is also provided.

The OS provides signals in two frequency bands to all users with suitable equipment free of charge. From FOC, it will offer a similar performance level to the modernized SPS GPS service, with a horizontal accuracy of order 2m (1σ) and a vertical accuracy of order 4m (1σ). It may be brought into operation in advance of the other services.

The SOL service uses the same signals as the OS, but adds signal integrity and authentication data, which validates that the Galileo signal is genuine, protecting against spoofing.

The CS will provide improved accuracy for those users willing to pay subscription charges. Encrypted signals in a third frequency band provide greater protection against interference and improved carrier-phase positioning (see Section 8.2.1). In addition, higher precision ephemeris and satellite clock parameters and local differential corrections may be broadcast. Commercial services may also be used to provide paying subscribers with a financially guaranteed service level.

The PRS is intended to provide high integrity, continuity, and some interference resistance to trusted subscribers in EU member states, such as emergency services and security services. However, the accuracy will be slightly poorer than that obtained from the open service at 3m (1σ) horizontally and 6m (1σ) vertically. It is not a specifically military service, but is operated in a similar manner to the GPS PPS, with dedicated spectrally separated and encrypted signals in two frequency bands. Only limited information about the PRS is available publicly for security reasons.

With the broadcasting of integrity alerts and differential corrections, Galileo effectively has built in SBAS. The term *Galileo local component* has been adopted to encompass GBAS and other locally based enhancements to the global Galileo service. The remainder of this section describes the Galileo space and control segments, signals, and navigation data messages.

6.4.1 Space and Ground Segments

Galileo will have a nominal constellation of 27 satellites. The orbital radius is 29,600 km (23,200 km above the Earth's surface) with a period of 14 hours, 5 minutes, giving 1.7 orbits per sidereal day and a ground-track repeat period of 10 sidereal days. The satellites will be distributed between three orbital planes, separated by 120° of longitude and nominally inclined at 56° to the equator. Figure 6.13 illustrates this. Each plane will contain nine satellites, nominally with an equally spacing of 40° with a tolerance of $\pm 2^\circ$. Spare satellites will be kept in each

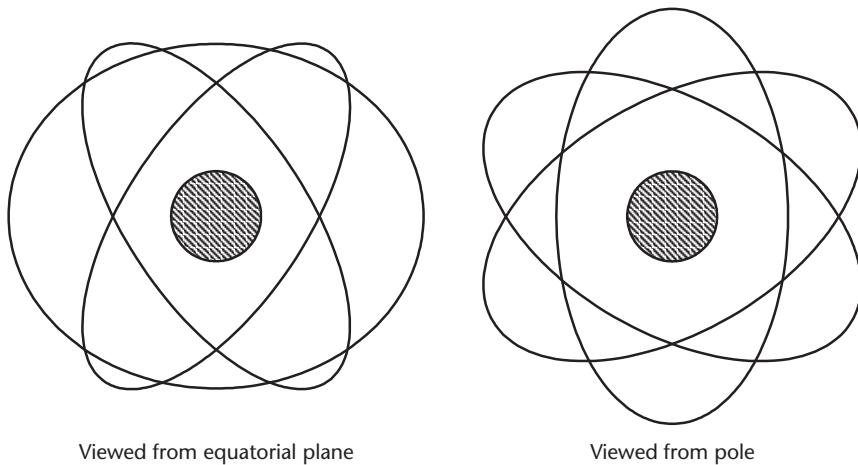


Figure 6.13 Galileo satellite orbits.

orbital plane, but will not be brought into operation until a satellite needs replacing [41]. The satellite hardware is described in [40].

The Galileo ground segment is partitioned into two separate systems, a ground control segment (GCS) and ground mission segment (GMS). Essentially, the GCS controls the satellite hardware, while the GMS controls the navigation service, including generation of the navigation data message content and integrity alerts. The GCS control center is located near Munich in Germany, while the GMS control center is at Fucino Space Center in Italy. Backup control centers for both the GCS and GMS will be located in Spain. The GCS will use a network of five TT&C stations. The GMS will use a separate network of 10 uplink stations, some of which are colocated with TT&C stations, and a global network of about 30 Galileo sensor stations (GSSs) to monitor the Galileo signals. For the IOV phase, there is 1 control center, 2 TT&C stations, 5 GMS uplink stations, and 18 GSSs. Batch processing over a 10-minute cycle is used to determine the precise orbit and clock offset of each satellite, while a separate integrity processing function (IPF) will provide instant alerts of satellite signal failures [41].

6.4.2 Signals

Galileo broadcasts 10 different navigation signals across three frequency bands: E5, E6, and E1-L1-E2 [8, 39]. The E5 band is 92.07 MHz (90×1.023 MHz) wide and centered at 1,191.795 MHz. It is partitioned into E5a and E5b subbands, with carrier frequencies of 1,176.45 and 1,207.14 MHz, respectively. The E5a subband coincides with the GPS L5 band, while the E5b subband overlaps the GLONASS L3 band. The E6 and E1-L1-E2 bands are both 40.92 MHz wide and centered at 1278.75 and 1575.42 MHz, respectively. The Galileo E1-L1-E2 band coincides with the GPS L1 band, but is wider. It is commonly abbreviated to L1. Table 6.5 summarizes the signals, while Figure 6.14 illustrates their PSDs. Note that two different naming conventions were in use at the time of writing. The E6C, E6P, and L1P signals use encrypted ranging codes.

Table 6.5 Galileo Signal Properties

Signal	Band and Carrier Frequency (MHz)	Services	Modulation and Chipping Rate ($\times 1.023 \text{ Mchip s}^{-1}$)	Navigation Message Rate (symbol s^{-1})	Minimum Received Signal Power (dBW)
E5a-d/ E5a-I	E5a, 1176.45	OS, CS	BPSK 10	50	-158
E5a-p/ E5a-Q	E5a, 1176.45	OS, CS	BPSK 10	None	-158
E5b-d/ E5b-I	E5b, 1207.14	OS, SOL, CS	BPSK 10	250	-158
E5b-p/ E5b-Q	E5b, 1207.14	OS, SOL, CS	BPSK 10	None	-158
E6P/ E6-A	E6, 1278.75	PRS	BOC _c (10,5)	Note 2	-155
E6C-d/ E6-B	E6, 1278.75	CS	BPSK 5	1000	-158
E6C-p/ E6-C	E6, 1278.75	CS	BPSK 5	None	-158
L1P/ E1-A	L1, 1575.42	PRS	BOC _c (15,2.5)	Note 2	-157
L1F-d/ E1-B	L1, 1575.42	OS, SOL, CS	BOC _s (1,1) Note 1	250	-160
L1F-p/ E1-C	L1, 1575.42	OS, SOL, CS	BOC _s (1,1) Note 1	None	-160

Note 1: This provisional L1F modulation scheme was recently replaced by a version of the MBOC modulation recently adopted for the GPS L1C signal (see Section 6.2.2) [18].

Note 2: The PRS navigation message rates are not in the public domain.

Table 6.6 gives the code lengths and repetition intervals for the Galileo OS, SOL, and CS ranging codes [42]. Layered codes are generally used. The total code length for the navigation-data-message-modulated signals is set to the data symbol length. For the data-free, or pilot, signals, a 100-ms code repetition period is used to ensure that the code length is not less than the satellite-to-user distance. Different primary codes are used for the data and pilot signals.

The data components of the E5 signals, E5a-d and E5b-d, are broadcast in-phase with respect to the carriers, while the pilot components, E5a-p and E5b-p, are broadcast in quadrature (i.e., 90° out of phase). The signals are sometimes referred to as E5a-I, E5b-I, E5a-Q, and E5b-Q, respectively.

The Galileo satellites broadcast the E5a and E5b signals as a combined signal multiplex. This enables them to share a single transmitter and provides the users with the option of tracking a single wideband signal, instead of separate signals, to obtain more accurate pseudo-range measurements. However, because the E5a-d and E5b-d signals carry different navigation data messages, standard BOC modulation cannot be used. Instead, an alternate-binary-offset-carrier (AltBOC) modulation scheme has been developed with a 15.345-MHz subcarrier frequency and 10.23-Mchip s^{-1} spreading code chipping rate. This permits differentiation of the sidebands.

Each of the four components of the AltBOC signal has a signal modulation with a constant phase with respect to the E5a or E5b carrier. Therefore the modulation phases with respect to the E5 AltBOC carrier rotate by one cycle per subcarrier period, with the E5a and E5b components rotating in opposite directions. To combine these signals, the AltBOC subcarrier modulation must perform the phase rotation. An eight-phase subcarrier modulation is therefore applied, which changes eight times per subcarrier period, as opposed to twice for conventional BOC modulation. The wideband signal comprises 8PSK modulation, as opposed to QPSK, enabling the receiver to separate the different signal modulations, E5a data, E5b data, and the pilots. An intermodulation product term is added to maintain constant

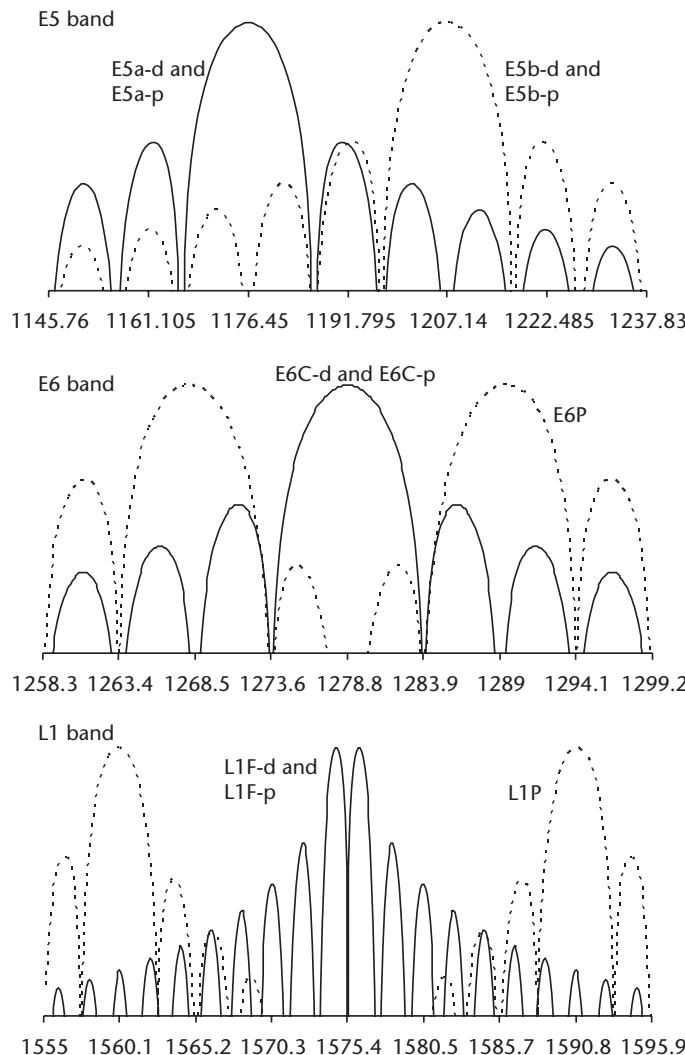


Figure 6.14 Galileo signal multiplex power spectral densities (log scale).

Table 6.6 Galileo OS, SOL, and CS Code Lengths

Signal	Primary Code		Secondary Code		Full Code	
	Length	Repetition Interval	Length	Length	Repetition Interval	
E5a-d	10,230	1 ms	20	204,600	20 ms	
E5b-d	10,230	1 ms	4	40,920	4 ms	
E5a-p, E5b-p	10,230	1 ms	100	1,023,000	100 ms	
E6C-d	5,115	1 ms	—	5,115	1 ms	
E6C-p	5,115	1 ms	100	511,500	100 ms	
L1F-d	4,092	4 ms	—	4,092	4 ms	
L1F-p	4,092	4 ms	25	102,300	100 ms	

signal modulus for transmission efficiency; the receiver does not need to decode this [40, 43, 44].

The three signals in each of the L1 and E6 bands are combined using coherent adaptive subcarrier modulation (CASM), also known as interplex. In the L1 band, the two L1F signals are modulated in-phase, and the L1P signal is in quadrature. In the E6 band, the two E6C signals are modulated in-phase, and the E6P signal is in quadrature. In both cases, an intermodulation product of the three multiplexed signals is added in the quadrature channel to maintain constant signal modulus [40, 43].

6.4.3 Navigation Data Messages

Galileo broadcasts four different data messages. The freely accessible (FNAV) message is carried on the E5a-d signal; the integrity (INAV) message is carried on the E5b-d and L1F-d signals; the commercial (CNAV) message is centered on the E6C-d signal and the government-access (GNAV) message is carried on both PRS signals. All Galileo navigation data messages use 1/2-rate FEC, except for the synchronization word, giving data rates of 25 bit s^{-1} for FNAV, 125 bit s^{-1} for INAV, and 500 bit s^{-1} for CNAV. All messages have a flexible frame structure [45].

Navigation data are broadcast on all message types except for CNAV. Ephemeris and almanac data are similar to that in the GPS legacy message, while the satellite clock parameters are at a higher resolution for both the transmitting satellite and the constellation. A common issue of data navigation (IODNav) integer is incremented when the ephemeris and clock data are updated, every 3 hours. There is also an issue of data almanac (IODA) integer. An 8-bit GPS-like health indicator is transmitted for each satellite, while the ionosphere propagation delay correction model uses three coefficients, rather than eight.

Galileo System Time (GST) is maintained within 50 ns of International Atomic Time (TAI). Like GPS time, GST is expressed in weeks and seconds, but with a roll over after 4,096 weeks (about 78 years). The Galileo data messages include both GST-UTC and Galileo-GPS time conversion data.

Integrity data, including three levels of integrity alert and authentication data, are transmitted on both the INAV and GNAV messages. The INAV messages on the E5b-d and L1F-d signals are staggered, enabling users that decode both to obtain integrity updates more frequently.

6.5 Regional Navigation Systems

This section describes the Chinese Beidou and Compass, Japanese QZSS, and Indian IRNSS systems. Compass, QZSS, and IRNSS will all broadcast GNSS signals, while Beidou operates on a different principle.

6.5.1 Beidou and Compass

The original Beidou (Big Dipper) navigation system uses a constellation of three geostationary satellites at longitudes of 80.2° , 110.4° , and 139.9° . The first satellite

was launched in late 2000 and the system became fully operational at the beginning of 2004. A good signal is available at longitudes between about 70° and 150° , spanning China and neighboring countries. The system is intended for road, rail, and maritime applications and operates completely independently of GNSS.

The signal geometry from geostationary satellites does not allow height and latitude to be observed separately so the user height is assumed to match that of the terrain or sea as appropriate. There is also a north-south hemisphere ambiguity in the latitude solution, so all users are assumed to be in the northern hemisphere. Furthermore, an accurate latitude solution is only available for users at latitudes above about 20° .

To avoid the need to synchronize the receiver clock, two-way ranging is used instead of passive ranging (see Section 1.3). Thus a two-dimensional navigation solution requires signals from only two of the satellites. The receiver records a short segment of the navigation signal and then transmits this to the satellite constellation at a fixed lag. This is then relayed to the control center, which computes the position solution and reports it to the user via satellite. The control center uses a terrain height database to obtain the correct latitude and longitude from the ranging measurement [46].

The Beidou satellites broadcast navigation signals at a carrier frequency of 2,491.75 MHz and the user equipment transmits back at 1,615.68 MHz. The code-chipping rate is $4.08 \text{ Mchip s}^{-1}$. The position accuracy within the area covered by the Beidou calibration station network is 20m (1σ) and the system can provide up to 540,000 position fixes per hour [47].

Beidou provides China with a satellite navigation system under its own control. However, the use of two-way ranging limits the duty cycle and number of users and introduces a lag of about a second. The Beidou geostationary satellites are also used as a platform for the SNAS augmentation system.

Starting with the launch of two further geostationary satellites in 2007, allocated to longitudes 58.8° and 160° , China began a program to develop a new satellite navigation system, known as Compass, using the Beidou satellites. This will add open-access GNSS-like signals and expand coverage.

A further three satellites will be in geosynchronous orbits, inclined to the equator at 55° and crossing it at a longitude of 118° [29]. These have a period of one sidereal day, but unlike geostationary orbits, they move with respect to the Earth's surface with varying latitude. This will allow three-dimensional positioning.

In addition, there are plans to expand Compass to provide global coverage by adding 27 MEO satellites with an orbital radius of 27,840 km and inclination of 55° . The first of these satellites was launched in 2007 [29].

Table 6.7 summarizes the properties of the Compass GNSS signals and codes known at the time of this writing [48]. A pilot signal and a navigation-data-modulated signal are broadcast on each of three frequencies, with layered codes used on the data-modulated signals. Additional $2.046 \text{ Mchip s}^{-1}$ signals may be broadcast at 1,589.742 MHz [29].

6.5.2 QZSS

The QZSS will provide positioning services primarily to in-car and personal receivers in Japan, though the navigation signals will be receivable across much of

Table 6.7 Compass GNSS Signal Properties and Code Lengths [48]

<i>Band and Carrier Frequency (MHz)</i>	<i>Modulation and Chipping Rate</i> $(\times 1,023 \text{ Mchip s}^{-1})$	<i>Navigation Message Rate</i> (symbol s^{-1})	<i>Length</i>	<i>Primary Code Repetition Interval</i>	<i>Secondary Code Length</i>	<i>Full Code Repetition Interval</i>
E5b, 1207.14	BPSK 10	None	Unknown	> 160 ms	—	> 160 ms
E5b, 1207.14	BPSK 2	50	2,046	1 ms	20	20 ms
E6, 1268.52	BPSK 10	None	Unknown	> 160 ms	—	> 160 ms
E6, 1268.52	BPSK 10	50	19,230	1 ms	20	20 ms
E2, 1561.098	BPSK 2	None	Unknown	> 400 ms	—	> 400 ms
E2, 1561.098	BPSK 2	50	2,046	1 ms	20	20 ms

East Asia and Oceania. The positioning service is intended to supplement GPS by increasing the number of satellites visible in urban canyons and mountainous regions. It will also provide a GPS differential corrections service to a higher resolution than MSAS. However, QZSS is not a true SBAS service, as it will not broadcast integrity information [49]. The first QZSS satellite is scheduled to launch in 2009.

The QZSS constellation will comprise three satellites in separate geosynchronous orbits, inclined to the equator at 45° . They are phased such that all satellites will share the same asymmetric figure-of-eight ground track over the Asia-Pacific region. This ensures that there is always at least one satellite over Japan at a high elevation angle [50].

QZSS will transmit navigation signals in four bands. Standard and SBAS versions of the C/A code and also the new GPS L1C signal will be broadcast in the L1 band and GPS-like L2C, L5I, and L5Q signals will be broadcast in the L2 and L5/E5a bands. The final signal, known as the L-band experimental (LEX) signal, shares the frequency and modulation of the Galileo E6C signals, but achieves a high navigation data rate at $250 \text{ symbol s}^{-1}$ by encoding 8 data bits onto each symbol [51].

6.5.3 IRNSS

IRNSS is intended to provide a fully independent GNSS service for India. It will be under the sole control of that country and is planned to be operational between 2009 and 2012, with the first satellite launch in 2007–2008. The service area will be from longitudes 40° to 140° and the accuracy within India will be about 20m (1σ) horizontally and vertically [52].

Three of the seven satellites will be geostationary and shared with the GAGAN SBAS system. The other four satellites will be divided between two geostationary orbits, inclined at 29° and crossing the equator at 55° and 112° . Two $10.23 \text{ Mchip s}^{-1}$ BPSK signals and a BOC_s(10,2) signal, all at 1,191.795 MHz, are currently proposed [29], with the possibility of further signals in the 2–4-GHz S-band.

6.6 GNSS Interoperability

At the time of this writing, GPS was the only fully operational global satellite navigation system. After 2011, GNSS is likely to comprise four fully operational

systems: GPS, GLONASS, Galileo, and Compass. This section discusses the relationship between GPS, GLONASS, and Galileo, covering frequency compatibility, competition for users, and multistandard user equipment.

6.6.1 Frequency Compatibility

The most fundamental requirement for coexistence of GPS, GLONASS, and Galileo is frequency compatibility; in other words, they must operate without interfering with each other. Figure 6.15 shows the frequency bands used by each system. GPS and GLONASS use completely separate bands, so there is little scope for interference between the satellites.

GPS and Galileo, however, share two frequency bands, L5/E5a and L1. In the L5/E5a band, both services broadcast open-access signals using CDMA with high code-chipping rates and moderate code lengths, so they can coexist without a problem. In the L1 band, however, there are two major issues. First, the GPS C/A code is very vulnerable to interference due to the short PRN code used. Second, the United States has a requirement to spectrally separate the GPS M code from all other GNSS signals.

The only overlap between the Galileo and GLONASS frequency bands is the Galileo L5b band and the GLONASS L3 band. However, with the two systems using high chipping-rate codes and different carrier frequencies, the potential for interference is minimal.

6.6.2 User Competition

GPS, GLONASS, and Galileo each have their exclusive user bases in terms of military users for GPS and GLONASS and PRS users for Galileo. Russian users will also have to use GLONASS signals, either exclusively or in conjunction with the other GNSS signals [35]. However, for many users, the three systems will be in competition. In this respect, GPS has a huge advantage in the form of a head start of nearly two decades. Many users only need a basic positioning service, for which GPS already meets their requirements. For differential and carrier-phase GNSS users, there is an extensive network of reference stations in place for GPS, which would have to replicated in order to provide the same service for Galileo and GLONASS. GPS differential corrections are free via SBAS, whereas users of Galileo's in-built differential corrections service may have to pay.

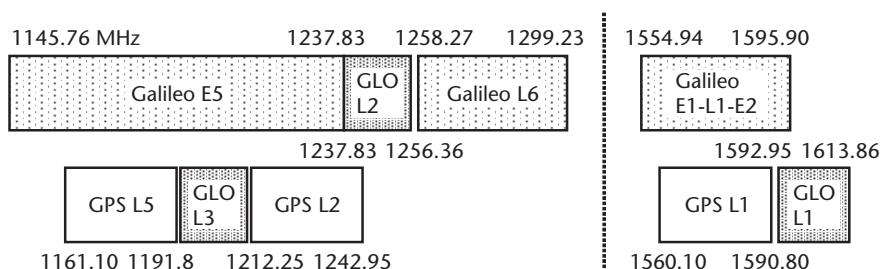


Figure 6.15 GPS, GLONASS, and Galileo frequency bands.

When it first enters full service, Galileo will have the advantage of being the first fully operational second-generation GNSS system, while GPS and GLONASS will still be part way through their modernization programs, with different satellites offering different services. Galileo will be the first to offer global integrity alerts and a high chipping-rate civil signal, providing improved multipath resistance (see Section 7.4.4), together with three-carrier positioning, at least for paying subscribers. In the longer term, GPS and GLONASS will offer open access signals in three bands, whereas Galileo will provide the highest precision ranging code on the AltBOC(15,10) signal in the E5 band. Galileo may also be the only system to offer a financially guaranteed service level. Realistically, the majority of Galileo and GLONASS users are likely to use these services in addition to GPS rather than instead of it.

6.6.3 Multistandard User Equipment

Making use of Galileo and/or GLONASS signals in addition to GPS brings a number of benefits to the user. More satellite signals brings more accurate positioning through averaging out of noise and error sources across more measurements (see Section 7.4.1), improved user-equipment-based integrity monitoring (see Section 15.4), and a much lower probability of having fewer than 4 satellites visible in difficult environments. It also provides insurance against complete failure of one of the navigation systems.

Multistandard receivers are, however, more complex. Operation on more frequencies and correlating a greater number of signals requires more hardware, while handling different types of signal and navigation message requires more complex software. Galileo was designed for interoperability with GPS, so it shares some of the carrier frequencies. GLONASS was designed independently of GPS and uses different frequencies, with use of FDMA bringing further hardware costs.

Another interoperability issue is reference datums and timebases. GPS uses the WGS 84 datum, whereas Galileo uses the GTRF datum, both based on the ITRF. WGS84, GTRF, and ITRF differ by only a few centimeters, so this is only an issue for high-precision users. GLONASS is moving from PZ-90 to an ITRF-based datum. All three systems use different timebases. Galileo time is based on TAI, whereas GPS and GLONASS times are based on, respectively, the U.S. and Russian versions of UTC. Although all three timebases are nominally synchronized, the differences are significant in GNSS-ranging terms. Galileo broadcasts GST-GPS time conversion data. However, no GLONASS-GPS time conversion data was broadcast at the time of writing, requiring the difference between U.S. and Russian UTC to be treated as an additional unknown in the navigation solution [38]. Plans to broadcast GST-GLONASS time conversion data from both satellite constellations were under consideration.

References

- [1] Dorsey, A. J., et al., "GPS System, Segments," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 67–112.

- [2] Spilker, J. J., Jr., and B. W. Parkinson, "Overview of GPS Operation and Design," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 29–55.
- [3] Misra, P., and P. Enge, *Global Positioning System Signals, Measurements, and Performance*, Lincoln, MA: Ganga-Jamuna Press, 2001.
- [4] Kaplan, E. D., et al., "Fundamentals of Satellite Navigation," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 21–65.
- [5] Klepczynski, W. J., "GPS for Precise Time and Time Interval Measurement," in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 483–500.
- [6] Spilker, J. J., Jr., "Signal Structure and Theoretical Performance," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson, and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 57–119.
- [7] Betz, J. W., "Binary Offset Carrier Modulation for Radionavigation," *Navigation: JION*, Vol. 48, No. 4, 2001, pp. 27–246.
- [8] Hein, G. W., et al., "Performance of Galileo L1 Signal Candidates," *Proc. ENC-GNSS 2004*, Rotterdam, the Netherlands, May 2004.
- [9] Parkinson, B. W., "Origins, Evolution, and Future of Satellite Navigation," *Journal of Guidance, Control and Dynamics*, Vol. 20. No. 1, 1997, pp. 11–25.
- [10] Parkinson, B. W., "Introduction and Heritage of NAVSTAR," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 3–28.
- [11] McDonald, K. D., "Early Development of the Global Positioning System," in *Galileo: Europe's Guiding Star*, W. Blanchard, (ed.), London, U.K.: Faircourt Ltd., 2006, pp. 114–128.
- [12] Van Graas, F., and M. S. Braasch, "Selective Availability," in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 601–621.
- [13] Aparicio, M., et al., "GPS Satellite and Payload," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 209–244.
- [14] Spilker, J. J., Jr., "Satellite Constellation and Geometric Dilution of Precision," in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 177–208.
- [15] Groves, P. D., "Principles of Integrated," Course Notes, QinetiQ Ltd., 2002.
- [16] Francisco, S. G., "GPS Operational Control Segment," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 435–466.
- [17] Creel, T., et al., "New, Improved GPS: The Legacy Accuracy Improvement Initiative," *GPS World*, March 2006, pp. 20–31.
- [18] Hein, G. W., et al., "MBOC: The New Optimized Spreading Modulation Recommended for GALILEO L1 OS and GPS L1C," *Proc. IEEE/ION PLANS*, San Diego, CA, April 2006, pp. 883–892.
- [19] Anon., *Navstar GPS Military-Unique Space Segment/User Segment Interfaces*, ICD-GPS-700, Revision A, ARINC, September 2004.
- [20] Fontana, R. D., et al., "The New L2 Civil Signal," *Proc. ION GPS 2001*, Salt Lake City, UT, September 2001, pp. 617–631.
- [21] Tran, M., "Performance Evaluation of the New GPS L5 and L2 Civil (L2C) Signals," *Navigation: JION*, Vol. 51, No. 3, 2004, pp. 199–212.
- [22] Barker, B. C., et al., "Overview of the GPS M Code Signal," *Proc. ION NTM*, Anaheim, CA, January 2000, pp. 542–549.

- [23] Dafesh, P., et al., "Description and Analysis of Time-Multiplexed M-Code Data," *Proc. ION 58th AM*, Albuquerque, NM, June 2002, pp. 598–611.
- [24] Spilker, J. J., Jr., and A. J. Van Dierendonck, "Proposed New L5 Civil GPS Codes," *Navigation: ION*, Vol. 48, No. 3, 2001, pp. 135–143.
- [25] Betz, J. W., et al., "Description of the L1C Signal," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 2080–2091.
- [26] Anon., *Navstar GPS Space Segment/Navigation User Interfaces*, IS-GPS-200, Revision D, ARINC, December 2004.
- [27] Spilker, J. J., Jr., "GPS Navigation Data," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 121–176.
- [28] Habereder, H., I. Schempp, and M. Bailey, "Performance Enhancements for the Next Phase of WAAS," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1350–1358.
- [29] Hein, G. W., et al., "Envisioning a Future GNSS System of Systems, Part 1," *Inside GNSS*, January/February 2007, pp. 58–67.
- [30] *Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment*, RTCA/DO229C, November 2001.
- [31] Enge, P. K., and A. J. Van Dierendonck, "Wide Area Augmentation System," in *Global Positioning System: Theory and Applications Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 117–142.
- [32] Braff, R., "Description of the FAA's Local Area Augmentation System (LAAS)," *Navigation: ION*, Vol. 44, No. 4, 1997, pp. 411–423.
- [33] Elrod, B. D., and A. J. Van Dierendonck, "Pseudolites," in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 51–79.
- [34] Crosby, G. K., et al., "A Ground-Based Regional Augmentation System (GRAS)—The Australian Proposal," *Proc. ION GPS 2000*, Salt Lake City, UT, September 2000, pp. 713–721.
- [35] Revnivykh, S., et al., "GLONASS Status, Performance and Perspectives," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005.
- [36] Feairheller, S., and R. Clark, "Other Satellite Navigation Systems," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 595–634.
- [37] Anon., "Radical Change in the Air for GLONASS," *GPS World*, February 2007, p. 14.
- [38] Daly, P., and P. N. Misra, "GPS and Global Navigation Satellite System," in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 243–272.
- [39] Ruiz, L., R. Crescenberri, and E. Breeuwer, "Galileo Services Definition and Navigation Performance," *Proc. ENC-GNSS 2004*, Rotterdam, the Netherlands, May 2004.
- [40] Falcone, M., P. Erhard, and G. W. Hein, "Galileo," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 559–594.
- [41] Dinwiddie, S. E., E. Breeuwer, and J. H. Hahn, "The Galileo System," *Proc. ENC-GNSS 2004*, Rotterdam, the Netherlands, May 2004.
- [42] Pratt, A. R., "Galileo Signal Structure," *Proc. NAV 05*, Royal Institute of Navigation, London, U.K., November 2005.
- [43] Kreher, J., *GALILEO Signals: RF Characteristics*, ICAO Navigation Systems Panel, Working Group of the Whole, Working Paper 36, October 2004.
- [44] Issler, J. -L., et al., "Spectral Measurements of GNSS Satellite Signals Need for Wide Transmitted Bands," *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 445–460.

- [45] *Galileo Open Service Signal in Space ICD (OS SIS ICS)*, Draft 0, Galileo Joint Undertaking, May 2006.
- [46] Forden, G., “The Military Capabilities and Implications of China’s Indigenous Satellite-Based Navigation System,” *Science and Global Security*, Vol. 12, 2004, pp. 219–250.
- [47] Bion, S., J. Jin, and Z. Fang, “The Beidou Satellite Positioning System and Its Positioning Accuracy,” *Navigation: JION*, Vol. 52, No. 3, 2005, pp. 123–129.
- [48] Grelier, T., et al., “Initial Observations and Analysis of Compass MEO Satellite Signals,” *Inside GNSS*, May–June 2007, pp. 39–43.
- [49] Petrovsky, I. G., “QZSS—Japan’s New Integrated Communication and Positioning Service for Mobile Users,” *GPS World*, June 2003, pp. 24–29.
- [50] Maeda, H., “QZSS Overview and Interoperability,” *Proc. ION GNSS 2005*, Long Beach, CA, September 2005.
- [51] Kogure, S., M. Sawabe, and M. Kishimoto, “Status of QZSS Navigation System in Japan,” *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 2092–2102.
- [52] Singh, A., and S. K. Saraswati, “India Heading for a Regional Navigation Satellite System,” *Coordinates*, November 2006, pp. 6–8.

Selected Bibliography

- El-Rabbany, A., *Introduction to GPS: The Global Positioning System*, 2nd ed., Norwood, MA: Artech House, 2006.
- Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, New York: Wiley, 2001.
- Hoffmann-Wellenhof, B., H. Lichtenegger, and J. Collins, *Global Positioning Systems*, 5th ed., Vienna, Austria: Springer, 2001.
- Van Dierendonck, A. J., “Satellite Radio Navigation,” in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 178–282.

Satellite Navigation Processing, Errors, and Geometry

This chapter describes how GNSS user equipment processes the signals from the satellites to obtain ranging measurements and then a navigation solution. It also reviews the error sources and describes the geometry of the navigation signals. It follows on from the fundamentals of satellite navigation described in Section 6.1.

Different authors describe GNSS user equipment architecture in different ways [1–4]. Here, it is divided into four functional blocks, as shown in Figure 7.1: the antenna, receiver hardware, ranging processor, and navigation processor. This approach splits up the signal processing, ranging, and navigation functions, matching the different INS/GNSS integration architectures described in Chapter 12.

Section 7.1 describes the geometry of satellite navigation, covering the satellite position and velocity, range and range rate, line of sight, azimuth and elevation, and navigation solution geometry. Section 7.2 describes the antenna and receiver hardware, with an emphasis on signal processing. Section 7.3 describes the ranging

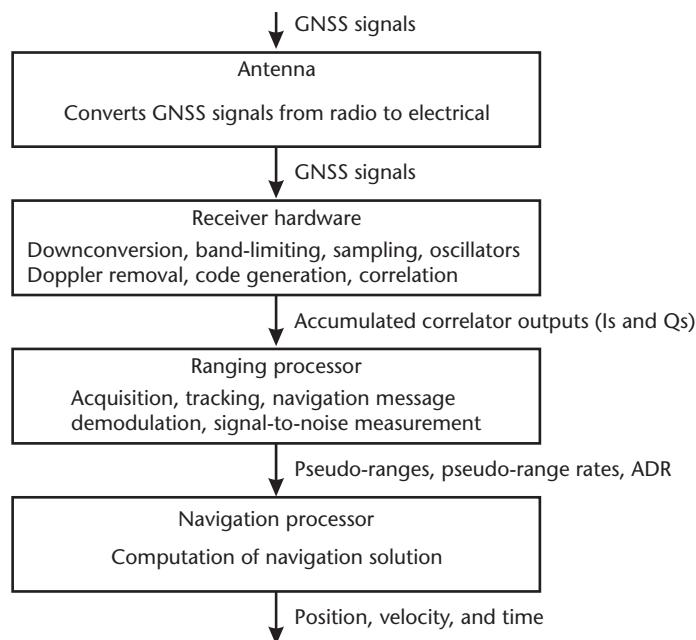


Figure 7.1 GNSS user equipment functional diagram.

processor, including acquisition, code and carrier tracking, lock detection, navigation message demodulation, signal-to-noise measurement, and generation of the pseudo-range, pseudo-range rate, and carrier-phase measurements. Section 7.4 discusses the error sources leading to ranging errors, including ephemeris and satellite clock errors, ionosphere and troposphere propagation errors, tracking errors, and multipath. The satellite clock, ionosphere, and troposphere errors are partially corrected by the user equipment, within either the ranging processor or the navigation processor. Finally, Section 7.5 describes the navigation processor, covering both single-point and filtered navigation solutions, discussing integrated navigation and tracking, and concluding with a discussion of navigation error budgets.

7.1 Satellite Navigation Geometry

This section describes the geometry of GNSS navigation. The calculation of the satellite positions and velocities from the information in the navigation data message is described first. This is followed by a discussion of range and range-rate computation, including the effect of Earth rotation and the impact of different errors at different processing stages. The direction of the satellite signal from the user antenna is then defined in terms of line-of-sight vector, elevation, and azimuth. Finally, the effect of signal geometry on navigation solution accuracy is discussed and the dilution of precision (DOP) defined.

7.1.1 Satellite Position and Velocity

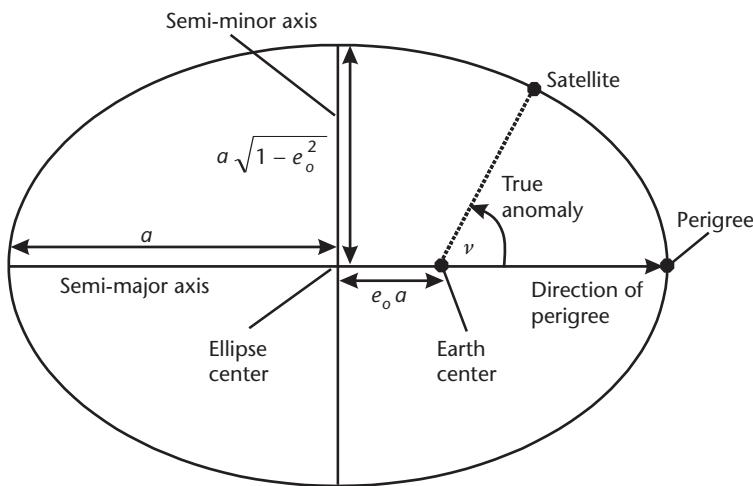
GPS and Galileo transmit satellite orbit data, known as the ephemeris, as a set of 16 quasi-Keplerian parameters. These are listed in Table 7.1, including the resolution (in terms of the least significant bit) applicable to the legacy GPS navigation data message [5]. Note that the ICDs use semicircles for many of the angular terms, whereas radians are used here. Two further parameters are used, both of which are considered constant: the Earth-rotation rate, ω_{ie} (see Section 2.3.4), and the Earth's gravitational constant, μ (see Section 2.3.5).

Although most GNSS satellite orbits are nominally circular, the eccentricity of the orbit must be accounted for in order to accurately determine the satellite position. A two-body Keplerian model is used as the baseline for the satellite motion. This assumes the satellite moves in an ellipse, subject to the gravitational force of a point source at one focus of the ellipse [3, 6]. Seven parameters are used to describe a pure Keplerian orbit: a reference time, t_{oe} , three parameters describing the satellite orbit within the orbital plane, and three parameters describing the orientation of that orbit with respect to the Earth.

Figure 7.2 illustrates the satellite motion within the orbital plane. The size of the orbit is defined by the length of the semi-major axis, a . This is simply the radius of the orbit at its largest point. The shape of the orbit is defined by the eccentricity, e_o , where the subscript o has been added to distinguish it from the eccentricity of the Earth's surface. The two foci are each located at a distance $e_o a$ along the semi-major axis from the center of the ellipse. The center of the

Table 7.1 GPS and Galileo Satellite Orbit Ephemeris Parameters

<i>Symbol</i>	<i>Description</i>	<i>Resolution (LSB)</i>
t_{oe}	Reference time of the ephemeris	16 seconds
M_0	Mean anomaly at the reference time	1.46×10^{-9} rad
e_o	Eccentricity of the orbit	1.16×10^{-10}
$a^{1/2}$	Square root of the semimajor axis	1.91×10^{-6} m $^{-0.5}$
Ω_0	Right ascension of ascending node of orbital plane at the weekly epoch	1.46×10^{-9} rad
i_0	Inclination angle at the reference time	1.46×10^{-9} rad
ω	Argument of perigee	1.46×10^{-9} rad
Δn	Mean motion difference from computed value	3.57×10^{-13} rad s $^{-1}$
$\dot{\Omega}_d$	Rate of change of longitude of the ascending node at the reference time	3.57×10^{-13} rad s $^{-1}$
\dot{i}_d	Rate of inclination	3.57×10^{-13} rad s $^{-1}$
C_{uc}	Amplitude of the cosine harmonic correction term to the argument of latitude	1.86×10^{-9} rad
C_{us}	Amplitude of the sine harmonic correction term to the argument of latitude	1.86×10^{-9} rad
C_{rc}	Amplitude of the cosine harmonic correction term to the orbit radius	0.0313 m
C_{rs}	Amplitude of the sine harmonic correction term to the orbit radius	0.0313 m
C_{ic}	Amplitude of the cosine harmonic correction term to the angle of inclination	1.86×10^{-9} rad
C_{is}	Amplitude of the sine harmonic correction term to the angle of inclination	1.86×10^{-9} rad

**Figure 7.2** Satellite motion within the orbital plane.

Earth is at one focus. The perigree is defined as the point of the orbit that approaches closest to the center of the Earth and is located along the semi-major axis. The direction of perigee points from the center of the ellipse to the perigree, via the center of the Earth. Finally, the location of the satellite within the orbit at the reference time is defined by the true anomaly, ν , which is the angle in the counter-clockwise direction from the direction of perigee to the line of sight from the

center of the Earth to the satellite. The true anomaly does not vary at a constant rate over the orbit, so GNSS satellites broadcast the mean anomaly, M , which does vary at a constant rate and from which the true anomaly can be calculated.

Figure 7.3 illustrates the orientation of the orbital plane with respect to the Earth. The inclination angle, i , is the angle subtended by the normal to the orbital plane and the polar axis of the Earth and takes values between 0° and 90° . The *ascending node* is the point where the orbit crosses the Earth's equatorial plane while the satellite is moving in the positive z direction of the ECI and ECEF frames (i.e., south to north). The *descending node* is where the orbit crosses the equatorial plane in the opposite direction. The ascending and descending nodes are nominally fixed in the ECI frame but move within the ECEF frame as the Earth rotates. Therefore, the longitude of the ascending node, Ω , also known as the right ascension, is defined at the reference time.

The final term determining the orbit is the orientation of the direction of perigee within the orbital plane. This is defined using the argument of perigee, ω , which is the angle in the counterclockwise direction from the direction of the ascending node from the center of the Earth to the direction of perigee. Figure 7.4 illustrates this, together with the axes of the orbital coordinate frame, which is denoted by the symbol o and centered at the Earth's center of mass, like the ECI and ECEF frames. The x -axis of the orbital frame defines the direction of the ascending node and lies in the Earth's equatorial plane. The z -axis defines the normal to the equatorial plane in the Earth's northern hemisphere, as shown in Figure 7.3, and the y -axis completes the right-handed orthogonal set.

GNSS satellites depart from pure Keplerian motion due to a combination of nonuniformity of the Earth's gravitational field, the gravitational fields of the Sun and Moon, solar radiation pressure, and other effects. These are approximated by the remaining ephemeris parameters: the mean motion correction, rates of change of the inclination and longitude of the ascending node, and the six harmonic correction terms.

Calculation of the satellite position comprises two steps: determination of the position within the orbital coordinate frame and transformation of this to the

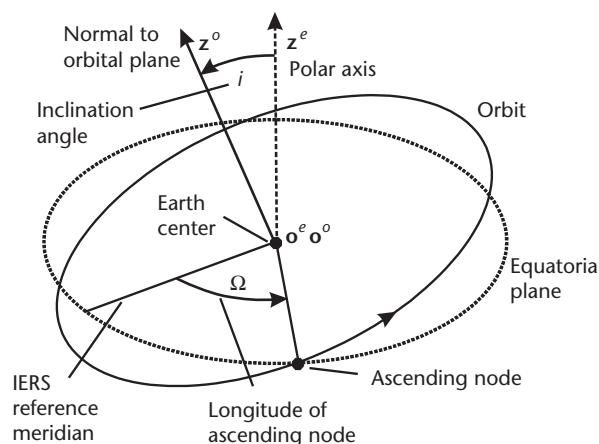


Figure 7.3 Orientation of the orbital plane with respect to the equatorial plane.

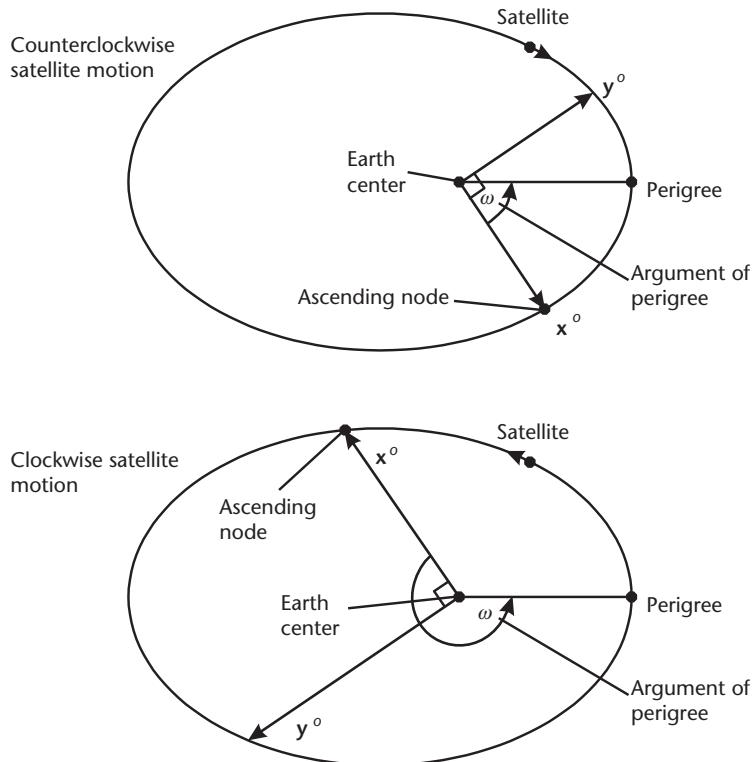


Figure 7.4 The argument of perigee and orbital coordinate frame axes.

ECEF or ECI frame as required. However, the time of signal transmission relative to the ephemeris reference time must first be determined:

$$\Delta t = t_{st} - t_{oe} \quad (7.1)$$

The GPS ephemeris reference time is transmitted relative to the start of the GPS week (see Section 6.2.3). Assuming the same for the time of signal transmission, t_{st} , it is sometimes necessary to apply a $\pm 604,800$ -second correction when the two times straddle the week crossover. This should be done where $|\Delta t| > 302,400$ seconds.

Next, the mean anomaly, M , is propagated to the signal transmission time using

$$M = M_0 + (\bar{\omega}_{is} + \Delta n) \Delta t \quad (7.2)$$

where the mean angular rate of the satellite's orbital motion, $\bar{\omega}_{is}$, is given by

$$\bar{\omega}_{is} = \sqrt{\mu/a^3} \quad (7.3)$$

The true anomaly, ν , is obtained from the mean anomaly via the eccentric anomaly, E . The eccentric anomaly, explained in [3, 7], is obtained using Kepler's equation:

$$M = E - e_o \sin E \quad (7.4)$$

which must be solved iteratively. A common numerical solution is¹

$$\begin{aligned} E_0 &= M + \frac{e_o \sin M}{1 - \sin(M + e_o) + \sin M} \\ E_i &= M + e_o \sin E_{i-1} \quad i = 1, 2, \dots, n \\ E &= E_n \end{aligned} \quad (7.5)$$

Performing 20 iterations (i.e., $n = 20$) should give centimetric accuracy, with 22 iterations giving millimetric accuracy.²

The true anomaly is then obtained from the eccentric anomaly using

$$\begin{aligned} \nu &= \text{arctan2}(\sin \nu_o, \cos \nu_o) \\ &= \text{arctan2}\left[\left(\frac{\sqrt{1 - e_o^2} \sin E}{1 - e_o \cos E}\right), \left(\frac{\cos E - e_o}{1 - e_o \cos E}\right)\right] \end{aligned} \quad (7.6)$$

where a four quadrant arctangent function must be used.

The position in the orbital coordinate frame may be expressed in polar coordinates comprising the radius, r_{os}^o , and argument of latitude, Φ , which is simply the sum of the argument of perigee and true anomaly, so

$$\Phi = \omega + \nu \quad (7.7)$$

The orbital radius varies as a function of the eccentric anomaly, while harmonic perturbations are applied to both terms, giving

$$\begin{aligned} r_{os}^o &= a(1 - e_o \cos E) + C_{rs} \sin 2\Phi + C_{rc} \cos 2\Phi \\ u_{os}^o &= \Phi + C_{us} \sin 2\Phi + C_{uc} \cos 2\Phi \end{aligned} \quad (7.8)$$

where u_{os}^o is the corrected argument of latitude.

The satellite position in the orbital frame is then

$$x_{os}^o = r_{os}^o \cos u_{os}^o, \quad y_{os}^o = r_{os}^o \sin u_{os}^o, \quad z_{os}^o = 0 \quad (7.9)$$

The position in the ECEF or ECI frame is obtained by applying a coordinate transformation matrix as the orbital frame has the same origin. Thus,

$$\mathbf{r}_{es}^e = \mathbf{C}_o^e \mathbf{r}_{os}^o, \quad \mathbf{r}_{is}^i = \mathbf{C}_o^i \mathbf{r}_{os}^o \quad (7.10)$$

The Euler rotation from the ECEF to the orbital frame comprises a yaw rotation through the longitude of the ascending node, Ω , followed by a roll rotation through the inclination angle, i . For GPS, the longitude of the ascending node is transmitted

at the week epoch, rather than the reference time, so its value at the time of signal transmission is

$$\Omega = \Omega_0 - \omega_{ie}(\Delta t + t_{oe}) + \dot{\Omega}_d \Delta t \quad (7.11)$$

while the inclination angle is corrected using

$$i = i_0 + \dot{i}_d \Delta t + C_{is} \sin 2\Phi + C_{ic} \cos 2\Phi \quad (7.12)$$

Applying (2.15) with $\psi_{eo} = \Omega$, $\phi_{eo} = i$, $\theta_{eo} = 0$ gives

$$C_o^e = \begin{pmatrix} \cos \Omega & -\cos i \sin \Omega & \sin i \sin \Omega \\ \sin \Omega & \cos i \cos \Omega & -\sin i \cos \Omega \\ 0 & \sin i & \cos i \end{pmatrix} \quad (7.13)$$

Thus, from (7.10), the ECEF-frame satellite position is

$$\mathbf{r}_{es}^e = \begin{pmatrix} x_{os}^o \cos \Omega - y_{os}^o \cos i \sin \Omega \\ x_{os}^o \sin \Omega + y_{os}^o \cos i \cos \Omega \\ y_{os}^o \sin i \end{pmatrix} \quad (7.14)$$

and, applying (2.94) and (2.95), the ECI-frame satellite position is

$$\mathbf{r}_{is}^i = \left\{ \begin{array}{l} x_{os}^o \cos [\Omega + \omega_{ie}(t_{st} - t_0)] - y_{os}^o \cos i \sin [\Omega + \omega_{ie}(t_{st} - t_0)] \\ x_{os}^o \sin [\Omega + \omega_{ie}(t_{st} - t_0)] + y_{os}^o \cos i \cos [\Omega + \omega_{ie}(t_{st} - t_0)] \\ y_{os}^o \sin i \end{array} \right\} \quad (7.15)$$

where t_0 is the time of coincidence of the ECI and ECEF-frame axes.

From (2.35), the satellite velocity is obtained simply by differentiating the position with respect to t_{st} . Differentiating (7.4) to (7.9) gives the satellite velocity in the orbital frame:

$$\dot{E} = \frac{\bar{\omega}_{is} + \Delta_n}{1 - e_o \cos E} \quad (7.16)$$

$$\dot{\Phi} = \frac{\sin \nu}{\sin E} \dot{E} \quad (7.17)$$

$$\dot{r}_{os}^o = (ae_o \sin E_k) \dot{E}_k + 2(C_{rs} \cos 2\Phi - C_{rc} \sin 2\Phi)\dot{\Phi} \quad (7.18)$$

$$\dot{u}_{os}^o = (1 + 2C_{us} \cos 2\Phi - 2C_{uc} \sin 2\Phi)\dot{\Phi}$$

$$\begin{aligned}\dot{x}_{os}^o &= \dot{r}_{os}^o \cos u_{os}^o - r_{os}^o \dot{u}_{os}^o \sin u_{os}^o \\ \dot{y}_{os}^o &= \dot{r}_{os}^o \sin u_{os}^o + r_{os}^o \dot{u}_{os}^o \cos u_{os}^o \\ \dot{z}_{os}^o &= 0\end{aligned}\quad (7.19)$$

Differentiating (7.11) and (7.12) gives

$$\dot{\Omega} = \dot{\Omega}_d - \omega_{ie} \quad (7.20)$$

$$\dot{i} = \dot{i}_d + 2(C_{is} \cos 2\Phi - C_{ic} \sin 2\Phi)\dot{\Phi} \quad (7.21)$$

Differentiating (7.14) and (7.15) then gives the ECEF and ECI-frame satellite velocities:

$$\begin{aligned}\mathbf{v}_{es}^e &= \begin{pmatrix} \dot{x}_{os}^o \cos \Omega - \dot{y}_{os}^o \cos i \sin \Omega + \dot{y}_{os}^o \sin i \sin \Omega \\ \dot{x}_{os}^o \sin \Omega + \dot{y}_{os}^o \cos i \cos \Omega - \dot{y}_{os}^o \sin i \cos \Omega \\ \dot{y}_{os}^o \sin i + \dot{y}_{os}^o \cos i \end{pmatrix} \\ &\quad + (\omega_{ie} - \dot{\Omega}_d) \begin{pmatrix} x_{os}^o \sin \Omega + y_{os}^o \cos i \cos \Omega \\ -x_{os}^o \cos \Omega + y_{os}^o \cos i \sin \Omega \\ 0 \end{pmatrix} \\ \mathbf{v}_{is}^i &= \left\{ \begin{array}{l} \dot{x}_{os}^o \cos [\Omega + \omega_{ie}(t_{st} - t_0)] - \dot{y}_{os}^o \cos i \sin [\Omega + \omega_{ie}(t_{st} - t_0)] + \dot{y}_{os}^o \sin i \sin [\Omega + \omega_{ie}(t_{st} - t_0)] \\ \dot{x}_{os}^o \sin [\Omega + \omega_{ie}(t_{st} - t_0)] + \dot{y}_{os}^o \cos i \cos [\Omega + \omega_{ie}(t_{st} - t_0)] - \dot{y}_{os}^o \sin i \cos [\Omega + \omega_{ie}(t_{st} - t_0)] \\ \dot{y}_{os}^o \sin i + \dot{y}_{os}^o \cos i \end{array} \right\} \\ &\quad - \dot{\Omega}_d \left\{ \begin{array}{l} x_{os}^o \sin [\Omega + \omega_{ie}(t_{st} - t_0)] + y_{os}^o \cos i \cos [\Omega + \omega_{ie}(t_{st} - t_0)] \\ -x_{os}^o \cos [\Omega + \omega_{ie}(t_{st} - t_0)] + y_{os}^o \cos i \sin [\Omega + \omega_{ie}(t_{st} - t_0)] \\ 0 \end{array} \right\} \quad (7.23)\end{aligned}$$

GLONASS simply transmits the ECEF-frame position, velocity, and acceleration at the ephemeris reference time. This is quicker to transmit and requires much less processing by the user. However, to maintain a given accuracy, the broadcast parameters must be updated more often. The satellite position and velocity at the time of signal transmission are obtained using

$$\mathbf{r}_{es}^e(t_{st}) = \mathbf{r}_{es}^e(t_{oe}) + \mathbf{v}_{es}^e(t_{oe})(t_{st} - t_{oe}) + \frac{1}{2} \mathbf{a}_{es}^e(t_{oe})(t_{st} - t_{oe})^2 \quad (7.24)$$

$$\mathbf{v}_{es}^e(t_{st}) = \mathbf{v}_{es}^e(t_{oe}) + \mathbf{a}_{es}^e(t_{oe})(t_{st} - t_{oe})$$

They may be transformed to the ECI frame using (2.94) to (2.96).

Table 7.2 presents the mean orbital radius, inertially referenced speed, and angular rate for the GPS, GLONASS, and Galileo constellations.

7.1.2 Range, Range Rate, and Line of Sight

The true range, ρ_T , is the distance between the satellite at the time of signal transmission, t_{st} , and the user's antenna at the time of signal arrival, t_{sa} . As Figure 7.5 shows, it is important to account for the signal transit time as the satellite-user distance generally changes over this interval, even where the user is stationary with respect to the Earth. The user equipment obtains pseudo-range measurements by multiplying its transit-time measurement by the speed of light. The speed of light in free space is only constant in an inertial frame; in a rotating frame, such as ECEF, it varies. Consequently, the true range calculation is simplest in the ECI frame. Thus, for satellite j ,

$$\rho_{Tj} = \left| \mathbf{r}_{is,j}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa}) \right| = \sqrt{(\mathbf{r}_{is,j}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa}))^T (\mathbf{r}_{is,j}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa}))} \quad (7.25)$$

As the user position is computed with respect to the Earth and the GPS ICD [5] gives formulas for computing the satellite position in ECEF coordinates, it is convenient to compute the range in the ECEF frame. However, this neglects the rotation of the Earth during the signal transit time, causing the range to be overestimated or underestimated as Figure 7.5 illustrates. At the equator, the range error can be up to 41m [8]. To compensate for this, a correction, $\delta\rho_{ie,j}$, known as the Sagnac or Earth-rotation correction, must be applied. Thus,

Table 7.2 GNSS Satellite Orbital Radii, Speeds, and Angular Rates

Constellation	GPS	GLONASS	Galileo
Mean orbital radius, \bar{r}_{es}	26,600 km	25,600 km	29,600 km
Mean satellite speed, \bar{v}_{is}	3,880 m s ⁻¹	4,200 m s ⁻¹	3,670 m s ⁻¹
Mean orbital angular rate, $\bar{\omega}_{is}$	1.46×10^{-4} rad s ⁻¹	1.64×10^{-4} rad s ⁻¹	1.24×10^{-4} rad s ⁻¹

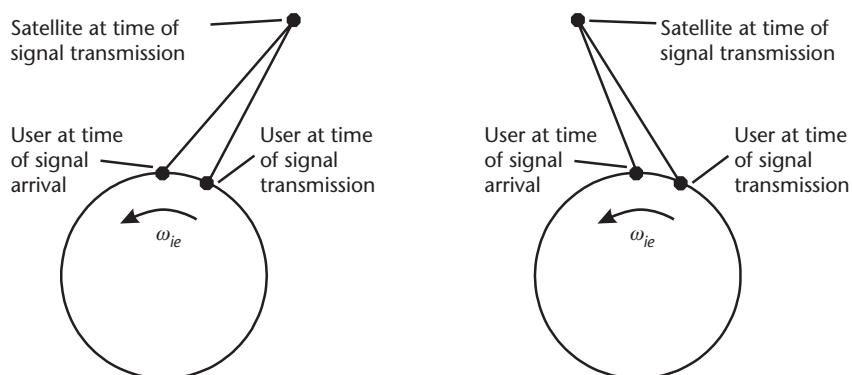


Figure 7.5 Effect of Earth rotation on range calculation (inertial frame perspective, user stationary with respect to the Earth).

$$\rho_{Tj} = \left| \mathbf{r}_{es,j}^e(t_{st,j}) - \mathbf{r}_{ea}^e(t_{sa}) \right| + \delta\rho_{ie,j} \quad (7.26)$$

However, computing the Sagnac correction exactly requires calculation of the ECI-frame satellite and user positions, so an approximation is generally used:

$$\delta\rho_{ie,j} \approx \frac{\omega_{ie}}{c} (y_{es,j}^e(t_{st,j})x_{ea}^e(t_{sa}) - x_{es,j}^e(t_{st,j})y_{ea}^e(t_{sa})) \quad (7.27)$$

The convenience of an ECEF-frame calculation can be combined with the accuracy of an ECI-frame calculation by aligning the ECI-frame axes with the ECEF-frame axes at the time of signal arrival or transmission [8]. From (2.94) and (2.95),

$$\mathbf{r}_{la}^I(t_{sa}) = \mathbf{r}_{ea}^e(t_{sa}) \quad \mathbf{r}_{ls,j}^I(t_{st,j}) = \mathbf{C}_e^I(t_{st,j}) \mathbf{r}_{es,j}^e(t_{st,j}) \quad (7.28)$$

where I denotes the ECI frame synchronized with the ECEF frame at the time of signal arrival and

$$\mathbf{C}_e^I(t) = \begin{pmatrix} \cos \omega_{ie}(t - t_{sa}) & -\sin \omega_{ie}(t - t_{sa}) & 0 \\ \sin \omega_{ie}(t - t_{sa}) & \cos \omega_{ie}(t - t_{sa}) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (7.29)$$

The range is then given by

$$\rho_{Tj} = \left| \mathbf{C}_e^I(t_{st,j}) \mathbf{r}_{es,j}^e(t_{st,j}) - \mathbf{r}_{ea}^e(t_{sa}) \right| \quad (7.30)$$

The small angle approximation may be applied to the rotation of the Earth during the signal transit time. Therefore, applying (6.1),

$$\mathbf{C}_e^I(t_{st,j}) \approx \begin{pmatrix} 1 & \omega_{ie}(t_{sa} - t_{st,j}) & 0 \\ -\omega_{ie}(t_{sa} - t_{st,j}) & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & \omega_{ie}\rho_{Tj}/c & 0 \\ -\omega_{ie}\rho_{Tj}/c & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (7.31)$$

The direction from which a satellite signal arrives at the user antenna may be described by a unit vector. The unit vector describing the direction of the origin of frame α with respect to the origin of frame β , resolved about the axes of frame γ , is denoted $\mathbf{u}_{\beta\alpha}^\gamma$ (some authors use \mathbf{l} or \mathbf{e}). Unit vectors have the property

$$\mathbf{u}_{\beta\alpha}^\gamma {}^T \mathbf{u}_{\beta\alpha}^\gamma \equiv \mathbf{u}_{\beta\alpha}^\gamma \cdot \mathbf{u}_{\beta\alpha}^\gamma = 1 \quad (7.32)$$

and the resolving axes are transformed using a coordinate transformation matrix:

$$\mathbf{u}_{\beta\alpha}^\delta = \mathbf{C}_\gamma^\delta \mathbf{u}_{\beta\alpha}^\gamma \quad (7.33)$$

The line-of-sight unit vector from the user to satellite j , resolved about the ECI frame axes, is

$$\mathbf{u}_{as,j}^i = \frac{\mathbf{r}_{is,j}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa})}{\left| \mathbf{r}_{is,j}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa}) \right|} = \frac{\mathbf{r}_{is,j}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa})}{\rho_{Tj}} \quad (7.34)$$

The ECEF-frame line-of-sight vector is

$$\mathbf{u}_{as,j}^e = \mathbf{C}_i^e(t_{sa}) \mathbf{u}_{as,j}^i \approx \frac{\mathbf{r}_{es,j}^e(t_{st,j}) - \mathbf{r}_{ea}^e(t_{sa})}{\left| \mathbf{r}_{es,j}^e(t_{st,j}) - \mathbf{r}_{ea}^e(t_{sa}) \right|} \quad (7.35)$$

The range rate is the rate of change of the range. Differentiating (7.25),

$$\dot{\rho}_{Tj} = \frac{(\mathbf{r}_{is,j}^i(t_{st,j}) - \mathbf{r}_{ia}^i(t_{sa}))^\top (\dot{\mathbf{r}}_{is,j}^i(t_{st,j}) - \dot{\mathbf{r}}_{ia}^i(t_{sa}))}{\rho_{Tj}} \quad (7.36)$$

Thus, applying (7.34), the range rate is obtained by resolving the satellite–antenna velocity difference along the line-of-sight unit vector:

$$\dot{\rho}_{Tj} = \mathbf{u}_{as,j}^i{}^\top (\mathbf{v}_{is,j}^i(t_{st,j}) - \mathbf{v}_{ia}^i(t_{sa})) \quad (7.37)$$

Applying (2.96), the range rate may be obtained from the ECEF-frame velocities using

$$\dot{\rho}_{Tj} = \mathbf{u}_{as,j}^e{}^\top [\mathbf{C}_e^I(t_{st,j}) (\mathbf{v}_{es,j}^e(t_{st,j}) + \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{es,j}^e(t_{st,j})) - (\mathbf{v}_{ea}^e(t_{sa}) + \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{ea}^e(t_{sa}))] \quad (7.38)$$

or, from (7.26),

$$\dot{\rho}_{Tj} = \mathbf{u}_{as,j}^e{}^\top (\mathbf{v}_{es,j}^e(t_{st,j}) - \mathbf{v}_{ea}^e(t_{sa})) + \delta\dot{\rho}_{ie,j} \quad (7.39)$$

where the range-rate Sagnac correction is approximately

$$\delta\dot{\rho}_{ie,j} \approx \frac{\omega_{ie}}{c} \begin{pmatrix} v_{es,j,y}^e(t_{st,j}) x_{ea}^e(t_{sa}) + y_{es,j}^e(t_{st,j}) v_{ea,x}^e(t_{sa}) \\ -v_{es,j,x}^e(t_{st,j}) y_{ea}^e(t_{sa}) - x_{es,j}^e(t_{st,j}) v_{ea,y}^e(t_{sa}) \end{pmatrix} \quad (7.40)$$

Applying (7.39) without the Sagnac correction leads to a range-rate error of up to 2 mm s^{-1} .

The true range and range rate are only of academic interest. A number of different ranges, pseudo-ranges, and range rates apply at different stages of the GNSS processing chain. The effective range that would be measured if the receiver and satellite clocks were perfectly synchronized is longer than the true range due to the refraction of the signal by the ionosphere and troposphere. Furthermore, the receiver actually measures the pseudo-range, which is also perturbed by the

satellite and receiver clock errors as described in Section 6.1.2. The pseudo-range and pseudo-range rate measured by the user equipment for satellite j are given by

$$\begin{aligned}\rho_{Rj} &= \rho_{Tj} + \delta\rho_{ij} + \delta\rho_{tj} - \delta\rho_{sj} + \delta\rho_{rc} \\ \dot{\rho}_{Rj} &= \dot{\rho}_{Tj} + \delta\dot{\rho}_{ij} + \delta\dot{\rho}_{tj} - \delta\dot{\rho}_{sj} + \delta\dot{\rho}_{rc}\end{aligned}\quad (7.41)$$

where $\delta\rho_{ij}$ and $\delta\rho_{tj}$ are, respectively, the ionosphere and troposphere propagation errors (see Section 7.4.2), $\delta\rho_{sj}$ is the range error due to the satellite clock (see Section 7.4.1), $\delta\rho_{rc}$ the range error due to the receiver clock (see Section 7.2.2), and $\delta\dot{\rho}_{ij}$, $\delta\dot{\rho}_{tj}$, $\delta\dot{\rho}_{sj}$, and $\delta\dot{\rho}_{rc}$ are their range-rate counterparts.

The raw pseudo-range and pseudo-range-rate measurements made by the receiver incorporate additional errors:

$$\begin{aligned}\tilde{\rho}_{Rj} &= \rho_{Rj} + \delta\rho_{wj} + \delta\rho_{mj} \\ \tilde{\dot{\rho}}_{Rj} &= \dot{\rho}_{Rj} + \delta\dot{\rho}_{wj} + \delta\dot{\rho}_{mj}\end{aligned}\quad (7.42)$$

where $\delta\rho_{wj}$ and $\delta\dot{\rho}_{wj}$ are the tracking errors (Section 7.4.3) and $\delta\rho_{mj}$ and $\delta\dot{\rho}_{mj}$ are the errors due to multipath (Section 7.4.4).

The navigation processor uses pseudo-range and pseudo-range-rate measurements with corrections applied. These are

$$\begin{aligned}\tilde{\rho}_{Cj} &= \tilde{\rho}_{Rj} + \Delta\rho_{icj} + \Delta\rho_{tcj} + \Delta\rho_{scj} \\ \tilde{\dot{\rho}}_{Cj} &= \tilde{\dot{\rho}}_{Rj} + \Delta\dot{\rho}_{scj}\end{aligned}\quad (7.43)$$

where $\Delta\rho_{icj}$ and $\Delta\rho_{tcj}$ are, respectively, the ionosphere and troposphere corrections (see Section 7.4.2), and $\Delta\rho_{scj}$ and $\Delta\dot{\rho}_{scj}$ are the satellite clock corrections (see Section 7.4.1).

Finally, most navigation processors make use of an estimated pseudo-range and pseudo-range rate given by

$$\begin{aligned}\hat{\rho}_{Cj} &= \left| \hat{\mathbf{r}}_{is,j}^i(t_{st,j}) - \hat{\mathbf{r}}_{ia}^i(t_{sa}) \right| + \hat{\delta\rho}_{rc}(t_{sa}) \\ \hat{\dot{\rho}}_{Cj} &= \hat{\mathbf{u}}_{as,j}^{i,\top} (\hat{\mathbf{v}}_{is,j}^i(t_{st,j}) - \hat{\mathbf{v}}_{ia}^i(t_{sa})) + \hat{\delta\dot{\rho}}_{rc}(t_{sa})\end{aligned}\quad (7.44)$$

where $\hat{\mathbf{r}}_{is,j}^i$ and $\hat{\mathbf{v}}_{is,j}^i$ are the estimated satellite position and velocity, obtained from the navigation data message, $\hat{\mathbf{r}}_{ia}^i$ and $\hat{\mathbf{v}}_{ia}^i$ are the navigation processor's estimates of the user antenna position and velocity, $\hat{\delta\rho}_{rc}$ and $\hat{\delta\dot{\rho}}_{rc}$ are the estimates of the receiver clock offset and drift, and $\hat{\mathbf{u}}_{as,j}^i$ is the line-of-sight vector obtained from the estimated satellite and user positions. The estimated range and range-rate are

$$\begin{aligned}\hat{\rho}_{Tj} &= \left| \hat{\mathbf{r}}_{is,j}^i(t_{st,j}) - \hat{\mathbf{r}}_{ia}^i(t_{sa}) \right| = \hat{\rho}_{Cj} - \hat{\delta\rho}_{rc}(t_{sa}) \\ \hat{\dot{\rho}}_{Tj} &= \hat{\mathbf{u}}_{as,j}^{i,\top} (\hat{\mathbf{v}}_{is,j}^i(t_{st,j}) - \hat{\mathbf{v}}_{ia}^i(t_{sa})) = \hat{\dot{\rho}}_{Cj} - \hat{\delta\dot{\rho}}_{rc}(t_{sa})\end{aligned}\quad (7.45)$$

The errors in the estimated range and range rate are

$$\begin{aligned}\hat{\rho}_{Tj} - \rho_{Tj} &= \delta\rho_{ej} - \mathbf{u}_{as,j}^i {}^\text{T} \delta\mathbf{r}_{ia}^i(t_{sa}) \\ \hat{\dot{\rho}}_{Tj} - \dot{\rho}_{Tj} &= \delta\dot{\rho}_{ej} - \mathbf{u}_{as,j}^i {}^\text{T} \delta\mathbf{v}_{ia}^i(t_{sa})\end{aligned}\quad (7.46)$$

where $\delta\rho_{ej}$ and $\delta\dot{\rho}_{ej}$ are the range and range-rate errors due to the ephemeris data in the navigation message (see Section 7.4.1), while $\delta\mathbf{r}_{ia}^i$ and $\delta\mathbf{v}_{ia}^i$ are the errors in the user position and velocity solution.

7.1.3 Elevation and Azimuth

The direction of a GNSS satellite from the user antenna is commonly described by an elevation, $\theta_{nu,j}$, and azimuth, $\psi_{nu,j}$. These angles define the orientation of the line-of-sight vector with respect to the north, east, and down axes of the local navigation frame, as shown in Figure 7.6, and correspond to the elevation and azimuth angles used to describe the attitude of a body (see Section 2.2.1). They may be obtained from the line-of-sight vector in the local navigation frame, $\mathbf{u}_{as,j}^n = (u_{as,j,N}^n, u_{as,j,E}^n, u_{as,j,D}^n)$, using

$$\theta_{nu,j} = -\arcsin(u_{as,j,D}^n), \quad \psi_{nu,j} = \arctan2(u_{as,j,E}^n, u_{as,j,N}^n) \quad (7.47)$$

where a four-quadrant arctangent function must be used. The reverse transformation is

$$\mathbf{u}_{as,j}^n = \begin{pmatrix} \cos \theta_{nu,j} \cos \psi_{nu,j} \\ \cos \theta_{nu,j} \sin \psi_{nu,j} \\ -\sin \theta_{nu,j} \end{pmatrix} \quad (7.48)$$

The local navigation frame line-of-sight vector is transformed to and from its ECEF and ECI-frame counterparts using (7.33) and (2.99) or (2.101).

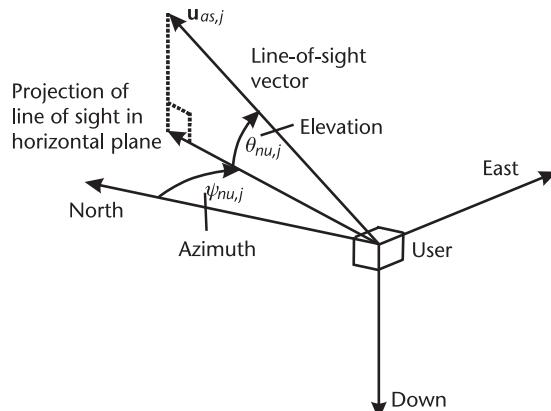


Figure 7.6 Satellite elevation and azimuth.

7.1.4 Signal Geometry and Navigation Solution Accuracy

The accuracy of a GNSS navigation solution depends not only on the accuracy of the ranging measurements, but also on the signal geometry. Figure 7.7 illustrates this for a simple two-dimensional ranging solution. The arcs show the mean and error bounds for each ranging measurement, while the shaded areas show the uncertainty bounds for the position solution and the arrows show the line-of-sight vectors from the user to the transmitters. The overall position error for a given ranging accuracy is minimized where the line-of-sight vectors are perpendicular.

The position information along a given axis obtainable from a given ranging signal is maximized when the angle between that axis and the signal line of sight is minimized. Therefore, the horizontal GNSS positioning accuracy is optimized where signals from low-elevation satellites are available and the line-of-sight vectors are evenly distributed in azimuth. Vertical accuracy is optimized where signals from higher elevation satellites are available. Figure 7.8 illustrates good and poor signal geometry.

Whereas the distribution of GNSS signals about azimuth is roughly uniform over time, the elevation distribution is biased toward lower elevations, particularly at high latitudes. Consequently, the horizontal position accuracy of GNSS is usually better than the vertical accuracy. The exception is in urban canyons and mountainous areas where many low-elevation signals are blocked.

The effect of signal geometry on the navigation solution is quantified using the dilution of precision (DOP) concept [9]. The uncertainty of each pseudo-range measurement, known as the user-equivalent range error (UERE), is σ_p . The DOP

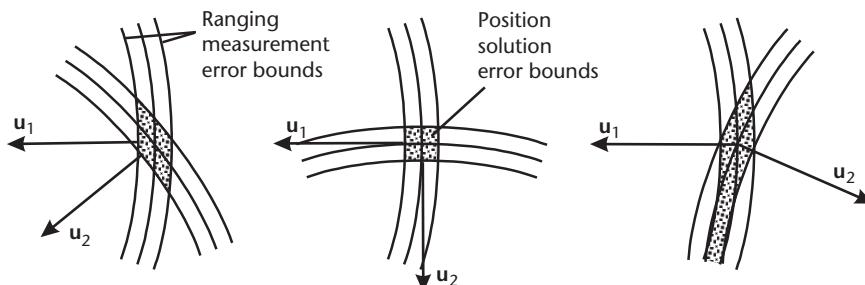


Figure 7.7 Effect of signal geometry on the position accuracy from two-dimensional ranging. (After: [3].)

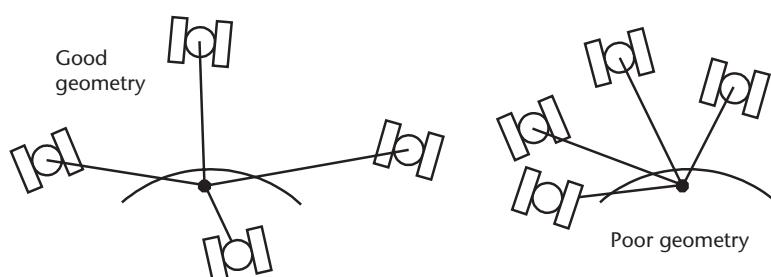


Figure 7.8 Examples of good and poor GNSS signal geometry.

is then used to relate the uncertainty of various parts of the navigation solution to the pseudo-range uncertainty using

$$\begin{aligned}\sigma_N &= D_N \sigma_\rho & \sigma_E &= D_E \sigma_\rho & \sigma_D &= D_V \sigma_\rho \\ \sigma_H &= D_H \sigma_\rho & \sigma_P &= D_P \sigma_\rho & \sigma_{rc} &= D_T \sigma_\rho \\ \sigma_G &= D_G \sigma_\rho\end{aligned}\quad (7.49)$$

where the various uncertainties and their DOPs are defined in Table 7.3.

Consider a GNSS receiver tracking signals from n satellites, each with a pseudo-range error $\delta\rho_j$. Using the line-of-sight unit vectors, the vector of pseudo-range errors, $\delta\rho$, may be expressed in terms of the navigation solution position error, $\delta\mathbf{r}_{ea}^n$, and residual receiver clock error, $\delta\delta\rho_{rc}$:

$$\begin{aligned}\delta\rho &= \begin{pmatrix} \delta\rho_1 \\ \delta\rho_2 \\ \vdots \\ \delta\rho_n \end{pmatrix} = \begin{pmatrix} -u_{as,1,N}^n & -u_{as,1,E}^n & -u_{as,1,D}^n & 1 \\ -u_{as,2,N}^n & -u_{as,2,E}^n & -u_{as,2,D}^n & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -u_{as,n,N}^n & -u_{as,n,E}^n & -u_{as,n,D}^n & 1 \end{pmatrix} \begin{pmatrix} \delta\mathbf{r}_{ea,N}^n \\ \delta\mathbf{r}_{ea,E}^n \\ \delta\mathbf{r}_{ea,D}^n \\ \delta\delta\rho_{rc} \end{pmatrix} \\ &= \mathbf{G}_n(\delta\mathbf{r}_{ea}^n, \delta\delta\rho_{rc})\end{aligned}\quad (7.50)$$

where \mathbf{G}_n is the local-navigation-frame geometry matrix. Squaring and taking expectations,

$$\mathbb{E}(\delta\rho\delta\rho^T) = \mathbf{G}_n \mathbb{E}[(\delta\mathbf{r}_{ea}^n, \delta\delta\rho_{rc})(\delta\mathbf{r}_{ea}^n, \delta\delta\rho_{rc})^T] \mathbf{G}_n^T \quad (7.51)$$

The error covariance matrix of the navigation solution is

$$\mathbf{P} = \mathbb{E}[(\delta\mathbf{r}_{ea}^n, \delta\delta\rho_{rc})(\delta\mathbf{r}_{ea}^n, \delta\delta\rho_{rc})^T] = \begin{pmatrix} \sigma_N^2 & P_{N,E} & P_{N,D} & P_{N,rc} \\ P_{E,N} & \sigma_E^2 & P_{E,D} & P_{E,rc} \\ P_{D,N} & P_{D,E} & \sigma_D^2 & P_{D,rc} \\ P_{rc,N} & P_{rc,E} & P_{rc,D} & \sigma_{rc}^2 \end{pmatrix} \quad (7.52)$$

Table 7.3 Uncertainties and Corresponding Dilutions of Precision

Uncertainty	Dilution of Precision
σ_N , north position	D_N , north dilution of precision
σ_E , east position	D_E , east dilution of precision
σ_D , vertical position	D_V , vertical dilution of precision (VDOP)
σ_H , horizontal position	D_H , horizontal dilution of precision (HDOP)
σ_P , overall position	D_P , position dilution of precision (PDOP)
σ_{rc} , receiver clock offset	D_T , time dilution of precision (TDOP)
σ_G , total position and clock	D_G , geometric dilution of precision (GDOP)

Assuming that the pseudo-range errors are independent and have the same uncertainties gives

$$\mathbb{E}(\delta\rho\delta\rho^T) = \mathbf{I}_n \sigma_\rho^2 \quad (7.53)$$

noting that, in reality, this does not apply to the ionosphere and troposphere propagation errors. Substituting (7.52) and (7.53) into (7.51) and rearranging gives

$$\mathbf{P} = \mathbf{G}_n^{-1} \mathbf{G}_n^{T-1} \sigma_\rho^2 = (\mathbf{G}_n^T \mathbf{G}_n)^{-1} \sigma_\rho^2 \quad (7.54)$$

From (7.49) and (7.52), the DOPs are then defined in terms of the geometry matrix by

$$\begin{pmatrix} D_N^2 & \cdot & \cdot & \cdot \\ \cdot & D_E^2 & \cdot & \cdot \\ \cdot & \cdot & D_V^2 & \cdot \\ \cdot & \cdot & \cdot & D_T^2 \end{pmatrix} = (\mathbf{G}_n^T \mathbf{G}_n)^{-1} \quad (7.55)$$

and

$$\begin{aligned} D_H &= \sqrt{D_N^2 + D_E^2} \\ D_P &= \sqrt{D_N^2 + D_E^2 + D_V^2} \\ D_G &= \sqrt{D_N^2 + D_E^2 + D_V^2 + D_{rc}^2} = \sqrt{\text{tr}[(\mathbf{G}_n^T \mathbf{G}_n)^{-1}]} \end{aligned} \quad (7.56)$$

From (7.50),

$$\mathbf{G}_n^T \mathbf{G}_n = \begin{pmatrix} g_{NN} & g_{NE} & g_{ND} & g_{NT} \\ g_{NE} & g_{EE} & g_{ED} & g_{ET} \\ g_{ND} & g_{ED} & g_{DD} & g_{DT} \\ g_{NT} & g_{ET} & g_{DT} & 1 \end{pmatrix} \quad (7.57)$$

where

$$\begin{aligned} g_{NN} &= \sum_{i=1}^n u_{as,i,N}^n & g_{NT} &= -\sum_{i=1}^n u_{as,i,N}^n & g_{NE} &= \sum_{i=1}^n u_{as,i,N}^n u_{as,i,E}^n \\ g_{EE} &= \sum_{i=1}^n u_{as,i,E}^n & g_{ET} &= -\sum_{i=1}^n u_{as,i,E}^n & g_{ND} &= \sum_{i=1}^n u_{as,i,N}^n u_{as,i,D}^n \\ g_{DD} &= \sum_{i=1}^n u_{as,i,D}^n & g_{DT} &= -\sum_{i=1}^n u_{as,i,D}^n & g_{ED} &= \sum_{i=1}^n u_{as,i,E}^n u_{as,i,D}^n \end{aligned} \quad (7.58)$$

As $\mathbf{G}_n^T \mathbf{G}_n$ is symmetric about the diagonal, the matrix inversion is simplified. Matrix inversion techniques are discussed in Section A.4 of Appendix A.

Table 7.4 gives average DOPs of a nominal 24-satellite GPS constellation at a range of latitudes, assuming signals are tracked from all satellites in view. Note that the VDOP is much larger than the HDOP, particularly in polar regions, even though the HDOP accounts for two axes. Overall performance is best in equatorial regions, though the difference is not great. Early GPS receivers could only track signals from four or five satellites at a time, and many used optimization of GDOP or PDOP to decide which satellites to select.

7.2 Receiver Hardware and Antenna

This section describes the hardware components of GNSS user equipment, shown in Figure 7.9. A brief discussion of antennas, the reference oscillator, and the receiver clock is followed by a description of the processing performed by the receiver hardware in the front-end and then the baseband signal processor. In the front-end, the GNSS signals are amplified, filtered, downconverted, and sampled.

Table 7.4 Average DOPs for a Nominal GPS Constellation and an All in View Receiver

Latitude	0°	30°	60°	90°
GDOP	1.78	1.92	1.84	2.09
PDOP	1.61	1.71	1.65	1.88
HDOP	0.80	0.93	0.88	0.75
VDOP	1.40	1.43	1.40	1.73
TDOP	0.76	0.88	0.80	0.90

Source: QinetiQ Ltd.

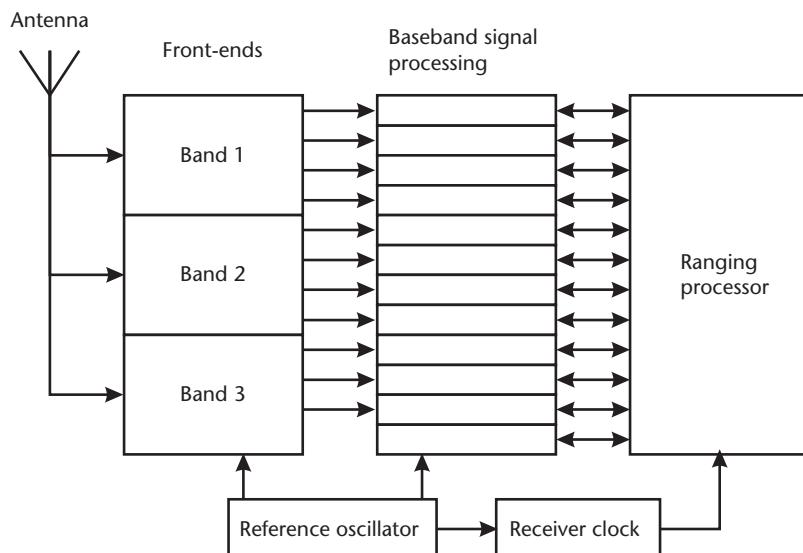


Figure 7.9 GNSS receiver hardware architecture.

In the baseband signal processor, the signals are correlated with internally generated code and carrier to produce the in-phase, I, and quadrature, Q, accumulated correlator outputs, which are provided to the ranging processor.

7.2.1 Antennas

A receiving antenna converts an electromagnetic signal into an electrical signal so that it may be processed by a radio receiver. A transmitting antenna performs the reverse operation. The gain of an antenna varies with frequency. Therefore, GNSS user equipment must incorporate an antenna that has peak sensitivity near to the carrier frequency of the signals processed by the receiver and sufficient bandwidth to pass those signals. Where the receiver processes signals in more than one frequency band, either the antenna must be sensitive in all of the bands required or a separate antenna for each band must be used. The antenna bandwidth should match or exceed the precorrelation bandwidth of the receiver (see Section 7.2.3).

A GNSS antenna should generally be sensitive to signals from all directions. A typical GNSS antenna has a gain of 2 to 4 dB for signals at normal incidence. This drops as the angle of incidence increases and is generally negative (in decibel terms) for angles of incidence greater than 75°. For a horizontally mounted antenna, a 75° incidence angle corresponds to a satellite signal at a 15° elevation angle. Some typical GPS antenna gain patterns are shown in [1, 2].

GNSS signals are transmitted with right-handed circular polarization (RHCP). On surface reflection, this is reversed to left-handed circular polarization (LHCP) or to elliptical polarization, which mixes RHCP and LHCP. Therefore, to minimize multipath problems (Section 7.4.4), the antenna should be sensitive only to RHCP signals.

For high-precision applications, it is important to know where the electrical phase center of the antenna is. This is the point in space for which the GNSS user equipment determines the navigation solution and does not necessarily coincide with the physical center of the antenna. For a given antenna, the phase center can vary with elevation, azimuth, and frequency.

Basic GNSS antennas come in a number of shapes and sizes [4]. Patch, or microstrip, antennas have the advantage of being low cost, flat, and rugged, but their polarization varies with the angle of incidence. Better performance can be obtained from a dome, blade, or helical (volute) antenna. More advanced antenna technology may be used to limit the effects of RF interference sources and/or multipath. This is discussed in Sections 8.3.1 and 8.4.1. For hand-held applications, the antenna is usually included with the user equipment, whereas for vehicles, a separate antenna is generally mounted on the vehicle body.

The cable between the antenna and the receiver imposes a common-mode lag on the incoming signal. However, the effects of antenna cable lag and receiver clock offset are indistinguishable, so the navigation processor simply accounts for the lag as part of its clock offset estimate. Signal attenuation in the antenna cable may be mitigated by including an amplifier in the antenna.

A full treatment of antennas may be found in suitable texts [10, 11].

7.2.2 Reference Oscillator and Receiver Clock

The timing in a GNSS receiver is controlled by the reference oscillator. This provides a frequency standard that drives both the receiver clock, which provides a time reference for the ranging and navigation processors, and the various oscillators used in the receiver front-end and baseband processor. Long-term errors and drift in the receiver's frequency standard are compensated in the navigation processor, so they do not present a problem, provided the frequency error is not large enough to disrupt the front-end. However, short-term variation in the oscillator frequency over the correlator coherent integration interval (see Section 7.2.4) and the time constant of the carrier tracking loop (see Section 7.3.3) can present a problem, particularly where the user equipment is optimized for poor signal-to-noise environments (see Section 8.3).

The vast majority of GNSS receivers use a quartz crystal oscillator (XO) as the frequency standard. The dominant source of error is variation in frequency with temperature and the oscillator frequency can vary by one part in 10^5 or 10^6 over typical operating temperature ranges [4].

Higher performance receivers use a temperature-compensated crystal oscillator (TCXO), costing tens of dollars or Euros. A TCXO uses a temperature sensor to vary the oscillator control voltage, stabilizing the frequency to within one part in 10^8 over a one-second interval. The frequency normally varies continuously, subject to quantization in the control process, but can experience sudden changes, known as microjumps [12].

An oven-controlled crystal oscillator (OCXO) uses an oven to maintain the oscillator at a fixed temperature. This achieves a frequency variation of about one part in 10^{11} over a second, with a frequency bias of one part in 10^8 . However, an OCXO is relatively large, consuming significant power and costing about a thousand Euros or dollars, so its use is restricted to specialized applications, such as survey receivers [3, 12].

Quartz oscillators also exhibit frequency errors proportional to the applied specific force. The coefficient of this g-dependent error is different for each of the three axes of the oscillator body. Frequency errors vary between one part in 10^{12} and one part in 10^8 per m s^{-2} of acceleration. Large g-dependent errors can disrupt carrier tracking in high-dynamics and high-vibration environments [13].

With a TCXO or OCXO, the range-rate bias due to the receiver clock drift, which is calibrated by the navigation processor, is of the order of 3 m s^{-1} , while with a basic XO, it can be up to 3000 m s^{-1} . The clock offset with a TCXO or OCXO can build up to several milliseconds, giving a range bias of several hundred kilometers if the receiver is left on standby for a few days. It can be reset from GNSS system time once the user equipment has decoded the navigation data message.

Reference stations used for wide area differential GNSS, signal monitoring, and the control of the GNSS systems themselves need to provide accurate measurements of the range errors. To do this, they require a precise time reference. Therefore, a reference station receiver uses a cesium or rubidium atomic clock instead of a crystal oscillator, giving a short-term stability of one part in 10^{11} and a long-term stability of one part in 10^{12} to 10^{13} [3].

7.2.3 Receiver Front-End

The receiver front-end processes the GNSS signals from the antenna in the analog domain, known as *signal conditioning*, and then digitizes the signal for output to the baseband signal processor [3, 14]. All signals in the same frequency band are processed together, while multiband receivers incorporate one front-end for each frequency band. Receivers for GLONASS, which uses FDMA, require a front-end for each channel. Figure 7.10 shows a typical front-end architecture, comprising an RF processing stage, followed by two intermediate frequency (IF) downconversion stages, and then the analog-to-digital converter (ADC) [4]. An alternative, direct digitization, approach is discussed in [15].

The carrier frequency is downconverted from the original L-band radio frequency to a lower IF to enable a lower sampling rate to be used, reducing the baseband processing load. Although two IF downconversion stages are common, some receivers employ a single stage, while others use more stages. At each stage, the incoming signal at a carrier frequency f_i is multiplied with a sinusoid of frequency f_o , produced by a local oscillator, driven by the receiver's reference oscillator. This produces two signals at carrier frequencies $|f_i - f_o|$ and $f_i + f_o$, each with the same modulation as the incoming signal. The higher-frequency signal is normally eliminated using a bandpass filter (BPF) [3].

The front-end must amplify the signal from the antenna by about seven orders of magnitude to interface with current ADCs. To prevent problems due to feedback, the amplification is distributed between the RF and IF processing stages [4]. The signal also undergoes bandpass filtering at each stage to limit out-of-band interference. The bandwidth of the conditioned signals entering the ADC is known as the *precorrelation bandwidth*. The minimum double-sided bandwidth required is about twice the chipping rate for a BPSK signal and $2(f_s + f_{co})$ for a BOC(f_s, f_{co}) signal (see Section 6.1.3). However, a wider precorrelation bandwidth sharpens the code correlation function (see Section 7.2.4.1), which can improve performance, particularly multipath mitigation (Section 8.4.2). The maximum useful precorrelation bandwidth is the transmission bandwidth of the GNSS satellite. Table 7.5 gives the minimum and maximum useful precorrelation bandwidths for the main GNSS signals (see Chapter 6).

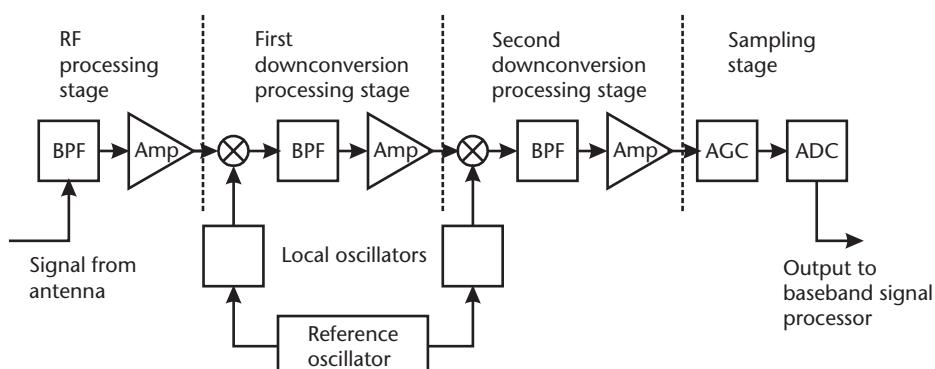


Figure 7.10 A typical GNSS receiver front-end architecture.

Table 7.5 Minimum and Maximum Receiver Precorrelation Bandwidths for GNSS Signals (Assuming Declared Transmission Bandwidth)

Signals	Minimum Prcorrelation Bandwidth (MHz)	Maximum Useful Prcorrelation Bandwidth (MHz)
GPS C/A and L2C	2.046	30.69
GPS P(Y) and L5	20.46	30.69
GPS M code	30.69	30.69
GPS L1C (BOC _s (1,1) component)	4.092	30.69
GLONASS C/A	1.022	10.22
GLONASS P	10.22	10.22
GLONASS L3	8.19	
Galileo E5a and E5b	20.46	40.92
Galileo E5 Alt BOC	51.15	92.07
Galileo L1F (BOC _s (1,1) component)	4.092	40.92
Galileo L1P	35.805	40.92
Galileo E6P	30.69	40.92
Galileo E6C	10.23	40.92

Assuming a BPSK signal for simplicity, the amplitude of a satellite signal received at the antenna phase center, neglecting band-limiting effects, is [3]

$$s_a(t_{sa}) = \sqrt{2P} C(t_{st}) D(t_{st}) \cos[2\pi(f_{ca} + \Delta f_{ca})t_{sa} + \phi_{ca}] \quad (7.59)$$

where P is the signal carrier power, C is the spreading code, D is the navigation data message, and ϕ_{ca} is the phase offset, as defined in Section 6.1.3. Here, f_{ca} is the transmitted carrier frequency, while Δf_{ca} is the Doppler shift due to the relative motion of the satellite and user-equipment antennas, t_{sa} is the time of signal arrival at the antenna, and t_{st} is the time of signal transmission. The Doppler shift is given by

$$\Delta f_{ca} = f_{ca} \left(\sqrt{\frac{1 - \dot{\rho}_R/c}{1 + \dot{\rho}_R/c}}^{-1} \right) \approx -\frac{f_{ca}}{c} \dot{\rho}_R \quad (7.60)$$

In this section, subscripts identifying the individual signals are omitted for brevity.

Following front-end processing, the signal amplitude, again neglecting band-limiting, is

$$s_{IF}(t_{sa}) = A_a C(t_{st}) D(t_{st}) \cos[2\pi(f_{IF} + \Delta f_{ca})t_{sa} + \phi_{ca} + \delta\phi_{IF}] \quad (7.61)$$

where A_a is the signal amplitude following amplification, f_{IF} is the final intermediate frequency, and $\delta\phi_{IF}$ is a phase shift common to all signals of the same type. Note that the magnitude of the Doppler shift is unchanged through the carrier-frequency downconversion process.

To prevent aliasing effects from disrupting carrier tracking, the ADC sampling rate must be at least twice the IF [3], while to prevent spectral foldover distorting the signal, the IF must exceed the single-sided precorrelation bandwidth. The sampling rate should be asynchronous with both the IF and the code-chipping

rate. This ensures that the samples vary in code and carrier phase, collectively encompassing the whole signal waveform.

Low-cost receivers use single-bit sampling. However, this reduces the effective signal to noise, known as an implementation loss. Better performance is obtained using a quantization level of two bits or more, together with an automatic gain control (AGC). The AGC varies the amplification of the input to the ADC to keep it matched to the dynamic range of the quantization process. As the noise dominates the signal prior to correlation, an AGC ensures a roughly constant noise standard deviation within the baseband signal processor and the measurements used by the ranging processor, while the signal level can vary. With a fast response rate, the AGC can be used to suppress pulsed interference [1]. This is particularly important in the L5/E5 band where interference from DME/TACAN and other sources may be a problem [16].

The sampling rate and quantization level of the ADC determines the processing power needed for the baseband signal processor. Consequently, there is a tradeoff between receiver performance and cost. It is important to match the IF to the precorrelation bandwidth to avoid a sampling rate higher than necessary.

7.2.4 Baseband Signal Processor

The baseband signal processor demodulates the sampled and conditioned GNSS signals from the receiver front-end by correlating them with internally generated replicas of the ranging (or spreading) code and carrier. The correlated samples are then summed and sent to the ranging processor, which controls the internally generated code and carrier.

Many authors class the ranging processor as part of the baseband processor, as the control loops used to acquire and track the signals span both the ranging-processor software and baseband signal-processor hardware. Here, the two are treated separately because the interface to the ranging processor traditionally marks the boundary between the hardware and software parts of current GNSS user equipment. In addition, advanced user equipment can combine the functions of the ranging and navigation processors (see Section 7.5.3).

The baseband signal processor is split into a series of parallel channels, one for each signal processed. Receivers typically have 10 or 12 channels per signal type, enabling signals to be tracked from all satellites in view. Figure 7.11 shows the architecture of a typical GNSS baseband signal processor channel. Advanced designs may implement more than six correlators to speed up acquisition (Section 7.3.1), mitigate multipath (Section 8.4.2), track BOC signals (Section 7.3.2), and mitigate poor signal-to-noise environments (Section 8.3).

In contemporary GNSS receivers, the baseband signal processing is implemented digitally in hardware. A number of experimental receivers, known as software receivers or software-defined receivers (SDRs), implement the baseband signal processing in software on a general-purpose processor [15, 17]. This conveys the advantage that the signal processor can be reconfigured to respond to different environments, such as high dynamics, poor signal to noise, and high multipath. The signal samples may also be stored, making it easier to resolve synchronization errors.

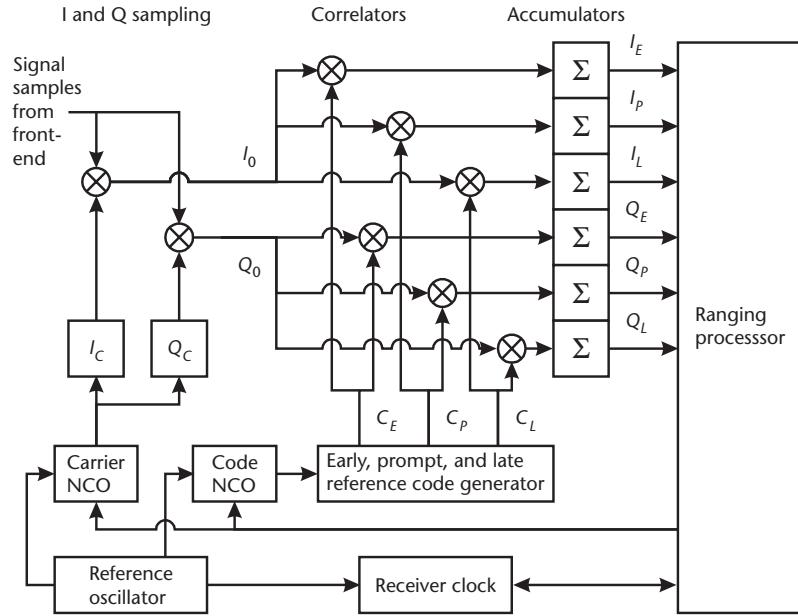


Figure 7.11 A GNSS baseband signal processor channel.

The baseband signal processing order varies according to the receiver design, but the outputs to the ranging processor are the same. Here, a typical approach is described.

The first stage in each channel is in-phase (I) and quadraphase (Q) sampling, also known as carrier wipe-off or Doppler wipe-off. Successive samples from the receiver front-end have a different carrier phase, so a summation of these will tend to zero, regardless of whether the signal and receiver-generated codes are aligned. The in-phase and quadraphase sampling transforms the precorrelation signal samples into two streams, I_0 and Q_0 , each with a nominally constant carrier phase. This is performed separately for each channel because the Doppler-shifted carrier frequency is different for each signal.

The I and Q samples are generated by multiplying the samples from the ADC, which comprise the wanted signal, s_{IF} , the unwanted signals on the same frequency, and noise, by in-phase and quadraphase samples of the receiver-generated carrier, I_C and Q_C , given by [3]

$$\begin{aligned} I_C(t_{sa}) &= 2 \cos[2\pi(f_{IF} + \tilde{\Delta f}_{ca})t_{sa} + \tilde{\phi}_{ca}] \\ Q_C(t_{sa}) &= 2 \sin[2\pi(f_{IF} + \tilde{\Delta f}_{ca})t_{sa} + \tilde{\phi}_{ca}] \end{aligned} \quad (7.62)$$

where $\tilde{\Delta f}_{ca}$ is the ranging processor's measurement of the Doppler shift and $\tilde{\phi}_{ca}$ is its measurement of the carrier phase offset after front-end processing. Again, the subscripts denoting the receiver channel are omitted for brevity. Approximate sine and cosine functions reduce the processor load [14]. The frequency $f_{IF} + \tilde{\Delta f}_{ca}$ is generated by the carrier numerically controlled oscillator (NCO), also known as a digitally controlled oscillator (DCO), which is driven by the reference oscillator and controlled by the ranging processor. Applying (7.61) to (7.62) and

neglecting the component at frequency $2f_{IF}$ and other harmonics, which are filtered out, the in-phase and quadraphase signal samples are then

$$\begin{aligned} I_0(t_{sa}) &= 2A_0 \cos[2\pi(\Delta f_{ca} - \tilde{\Delta f}_{ca})t_{sa} + \phi_{ca} + \delta\phi_{IF} - \tilde{\phi}_{ca}] + w_{I0}(t_{sa}) \quad (7.63) \\ Q_0(t_{sa}) &= 2A_0 \sin[2\pi(\Delta f_{ca} - \tilde{\Delta f}_{ca})t_{sa} + \phi_{ca} + \delta\phi_{IF} - \tilde{\phi}_{ca}] + w_{Q0}(t_{sa}) \end{aligned}$$

where A_0 is the signal amplitude following the AGC and ADC, and w_{I0} and w_{Q0} represent the noise from the receiver, RF interference, and the other satellite signals. Where the ranging processor's carrier phase and frequency estimates are correct, all of the signal power is in the in-phase samples. If the phase estimate is out by 90° , the signal power is in the quadraphase samples. Where there is an error in the frequency estimate, the signal power oscillates between the in-phase and quadraphase samples; the larger the frequency error, the shorter the period of oscillation.

The next stage of baseband signal processing is the code correlation. This comprises the multiplication of the precorrelation signal samples, I_0 and Q_0 , with the early prompt and late reference codes, given by

$$\begin{aligned} C_E(t_{sa}) &= C(\tilde{t}_{st} + d/2f_{co}) \\ C_P(t_{sa}) &= C(\tilde{t}_{st}) \\ C_L(t_{sa}) &= C(\tilde{t}_{st} - d/2f_{co}) \end{aligned} \quad (7.64)$$

where \tilde{t}_{st} is the ranging processor's measurement of the time of signal transmission and d is the code-phase offset in chips between the early and late reference signals. The prompt reference signal phase is halfway between the other two. The early-late correlator spacing varies between 0.05 and 1 chip, depending on the type of signal and the receiver design. The code correlation is also known as code wipeoff. The phase of the reference code generator is the integral of the code NCO output. This is driven by the reference oscillator and controlled by the ranging processor.

The correlator outputs are accumulated over an interval, τ_a , of at least 1 ms and then sent to the ranging processor. Although the accumulation is strictly a summation, there are sufficient samples to treat it as an integration for analytical purposes, and the accumulation is often known as *integrate and dump*. The early, prompt, and late in-phase and quadraphase accumulated correlator outputs are thus given by

$$\begin{aligned} I_E(t_{sa}) &= f_a \int_{t_{sa} - \tau_a}^{t_{sa}} I_0(t) C_E(t) dt, & Q_E(t_{sa}) &= f_a \int_{t_{sa} - \tau_a}^{t_{sa}} Q_0(t) C_E(t) dt \\ I_P(t_{sa}) &= f_a \int_{t_{sa} - \tau_a}^{t_{sa}} I_0(t) C_P(t) dt, & Q_P(t_{sa}) &= f_a \int_{t_{sa} - \tau_a}^{t_{sa}} Q_0(t) C_P(t) dt \quad (7.65) \\ I_L(t_{sa}) &= f_a \int_{t_{sa} - \tau_a}^{t_{sa}} I_0(t) C_L(t) dt, & Q_L(t_{sa}) &= f_a \int_{t_{sa} - \tau_a}^{t_{sa}} Q_0(t) C_L(t) dt \end{aligned}$$

where f_a is the ADC sampling frequency and noting that the time tag is applied to the end of the correlation interval here. These are commonly known simply as Is and Qs. For some methods of processing BOC signals, I_0 and Q_0 are also multiplied by a reference subcarrier function, $S_{E/P/L}$.

Substituting (7.63) and (7.64) into (7.65) and assuming an AGC is used, it may be shown that the accumulated correlator outputs are [1]

$$\begin{aligned} I_E(t_{sa}) &= \sigma_{IQ} \left[\sqrt{2(c/n_0)\tau_a} R(x - d/2) D(t_{st}) \operatorname{sinc}(\pi\delta f_{ca}\tau_a) \cos(\delta\phi_{ca}) + w_{IE}(t_{sa}) \right] \\ I_P(t_{sa}) &= \sigma_{IQ} \left[\sqrt{2(c/n_0)\tau_a} R(x) D(t_{st}) \operatorname{sinc}(\pi\delta f_{ca}\tau_a) \cos(\delta\phi_{ca}) + w_{IP}(t_{sa}) \right] \\ I_L(t_{sa}) &= \sigma_{IQ} \left[\sqrt{2(c/n_0)\tau_a} R(x + d/2) D(t_{st}) \operatorname{sinc}(\pi\delta f_{ca}\tau_a) \cos(\delta\phi_{ca}) + w_{IL}(t_{sa}) \right] \\ Q_E(t_{sa}) &= \sigma_{IQ} \left[\sqrt{2(c/n_0)\tau_a} R(x - d/2) D(t_{st}) \operatorname{sinc}(\pi\delta f_{ca}\tau_a) \sin(\delta\phi_{ca}) + w_{QE}(t_{sa}) \right] \\ Q_P(t_{sa}) &= \sigma_{IQ} \left[\sqrt{2(c/n_0)\tau_a} R(x) D(t_{st}) \operatorname{sinc}(\pi\delta f_{ca}\tau_a) \sin(\delta\phi_{ca}) + w_{QP}(t_{sa}) \right] \\ Q_L(t_{sa}) &= \sigma_{IQ} \left[\sqrt{2(c/n_0)\tau_a} R(x + d/2) D(t_{st}) \operatorname{sinc}(\pi\delta f_{ca}\tau_a) \sin(\delta\phi_{ca}) + w_{QL}(t_{sa}) \right] \end{aligned} \quad (7.66)$$

where σ_{IQ} is the noise standard deviation, c/n_0 is the carrier power to noise density, R is the code correlation function, x is the code tracking error in chips, δf_{ca} is the carrier frequency tracking error, $\delta\phi_{ca}$ is the carrier phase tracking error, and w_{IE} , w_{IP} , w_{IL} , w_{QE} , w_{QP} , and w_{QL} are the normalized I and Q noise terms. For the data-free, or pilot, GNSS signals, D is eliminated.

The tracking errors are defined as follows:

$$\begin{aligned} x &= (t_{st} - \tilde{t}_{st})f_{co} \\ \delta f_{ca} &= \Delta f_{ca} - \Delta \tilde{f}_{ca} \\ \delta\phi_{ca} &= \phi_{ca} + \delta\phi_{IF} - \tilde{\phi}_{ca} + (2\pi t_{sa} - \pi\tau_a)\delta f_{ca} \end{aligned} \quad (7.67)$$

From (6.1), (7.41), (7.42), and (7.60), the tracking errors may be expressed in terms of the pseudo-range and pseudo-range rate measurement errors by

$$\begin{aligned} x &= (\tilde{\rho}_R - \rho_R)f_{co}/c \\ \delta f_{ca} &= (\tilde{\rho}_R - \rho_R)f_{ca}/c \end{aligned} \quad (7.68)$$

The correlation function, carrier power to noise density, and noise properties are each described next, followed by discussions of the accumulation interval and signal multiplex processing.

7.2.4.1 Code Correlation Function

The code correlation function provides a measurement of the alignment between the signal and reference codes. Neglecting band-limiting effects, from (7.63) to (7.67), the correlation function for BPSK signals is

$$R_{BPSK}(x) = \frac{1}{\tau_a} \int_{t_{st} - \tau_a}^{t_{st}} C(t) C(t - x/f_{co}) dt \quad (7.69)$$

As stated in Section 6.3.1, the ranging code, C , takes values of ± 1 . Therefore, when the signal and reference codes are aligned, the correlation function $R(0)$ is unity. BPSK ranging codes are pseudo-random, so for tracking errors in excess of one code chip, there is a near-equal probability of the signal and reference code product being $+1$ or -1 at a given snapshot in time. Integrating this over the accumulation interval gives a nominal correlation function of zero. Where the tracking error is half a code chip, the signal and reference codes will match for half the accumulation interval and their product will average to near zero for the other half, giving a correlation function of $R(0.5) = 0.5$. Figure 6.5 illustrates this. More generally, the smaller the tracking error, the more the signal and reference will match. Thus, the correlation function is approximately

$$\begin{aligned} R_{BPSK}(x) &\approx 1 - |x| & |x| \leq 1 \\ &0 & |x| \geq 1 \end{aligned} \quad (7.70)$$

This is illustrated by Figure 7.12. In practice, the precorrelation band-limiting imposed at the satellite transmitter and the receiver front-end rounds the transitions of the signal code chips as illustrated by Figure 7.13. This occurs because a rectangular waveform is comprised of an infinite Fourier series of sinusoids, which is truncated by the band-limiting. The smoothing of the code chips results in a smoothing of the correlation function, as Figure 7.12 also shows.

One method of approximating the band-limited signal code is to replace the rectangular waveform with a trapezoidal waveform of rise time $\Delta x \approx 0.88f_{co}/B_{PC}$,

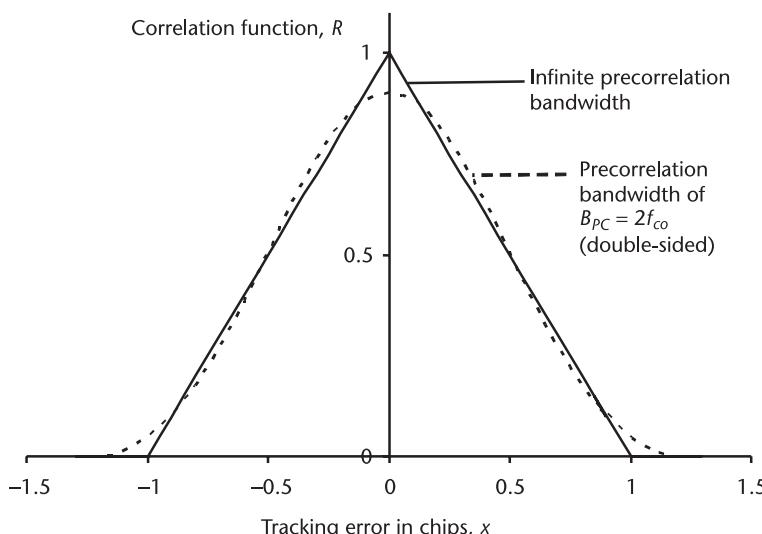


Figure 7.12 BPSK code correlation function for unlimited and band-limited signals.

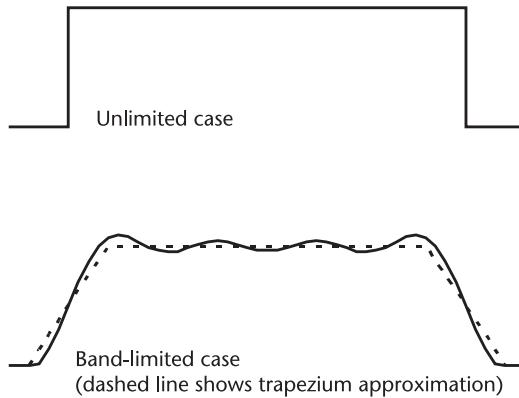


Figure 7.13 Comparison of unlimited and band-limited code chips.

where B_{PC} is the double-sided precorrelation bandwidth [18]. The correlation function under the trapezium approximation is [19]

$$\begin{aligned}
 R(x, \Delta x) &\approx 1 - \frac{\Delta x}{4} - \frac{x^2}{\Delta x} & 0 \leq |x| \leq \Delta x/2 \\
 &1 - |x| & \Delta x/2 \leq |x| \leq 1 - \Delta x/2 \\
 &\frac{1}{2\Delta x} - \left(\frac{|x|}{\Delta x} - \frac{1}{2}\right)\left(1 - \frac{|x|}{2} + \frac{\Delta x}{4}\right) & 1 - \Delta x/2 \leq |x| \leq 1 + \Delta x/2 \\
 &0 & 1 + \Delta x/2 \leq |x|
 \end{aligned} \tag{7.71}$$

The band-limited correlation function may be represented more precisely using Fourier analysis.

For code tracking errors greater than one chip, the auto-correlation function of a PRN code sequence is not exactly zero. Instead, it has noise-like behavior with a standard deviation of $1/\sqrt{n}$, where n is the number of code chips over the code repetition length or the accumulation interval, whichever is fewer [3]. The cross-correlation function between the reference code and a different PRN code of the same length also has a standard deviation of $1/\sqrt{n}$. The ranging codes used for GNSS signals are not randomly selected. For example, the GPS C/A codes are selected to limit the cross-correlation and minor auto-correlation peaks to $+0.064$ and -0.062 .

For BOC signals which are correlated with a reference spreading code and subcarrier function, S , together, the correlation function is more complex:

$$R_{BOC}(x) = \frac{1}{\tau_a} \int_{t_{st} - \tau_a}^{t_{st}} C(t) C(t - x/f_{co}) S(t) S(t - x/f_{co}) dt \tag{7.72}$$

The subcarrier function chips are shorter than the spreading-code chips and have a repetition period of less than or equal to the spreading-code chip size. Therefore, if the code tracking error is less than a spreading-code chip, but greater than a subcarrier-function chip, the reference and signal codes can be negatively correlated. Figure 7.14 shows the combined correlation functions for the main BOC GNSS signals [20].

A GNSS receiver may process only the upper or lower lobe of a BOC signal by adding $\pm f_s$ to $\Delta \tilde{f}_{ca}$ in (7.62) to shift the reference carrier frequency and then perform code correlation with the spreading code, C , but not the subcarrier function, S . This is useful for speeding up acquisition.

7.2.4.2 Signal to Noise

From (7.63) to (7.65), the postcorrelation signal amplitude is simply

$$A_C = f_a \tau_a A_0 \quad (7.73)$$

while the postcorrelation noise standard deviation is

$$\sigma_{IQ} = f_a \sqrt{E \left\{ \left[\int_{t_{sa}-\tau_a}^{t_{sa}} w_{I0}(t) C(t + t_{st} - t_{sa}) dt \right]^2 \right\}} \quad (7.74)$$

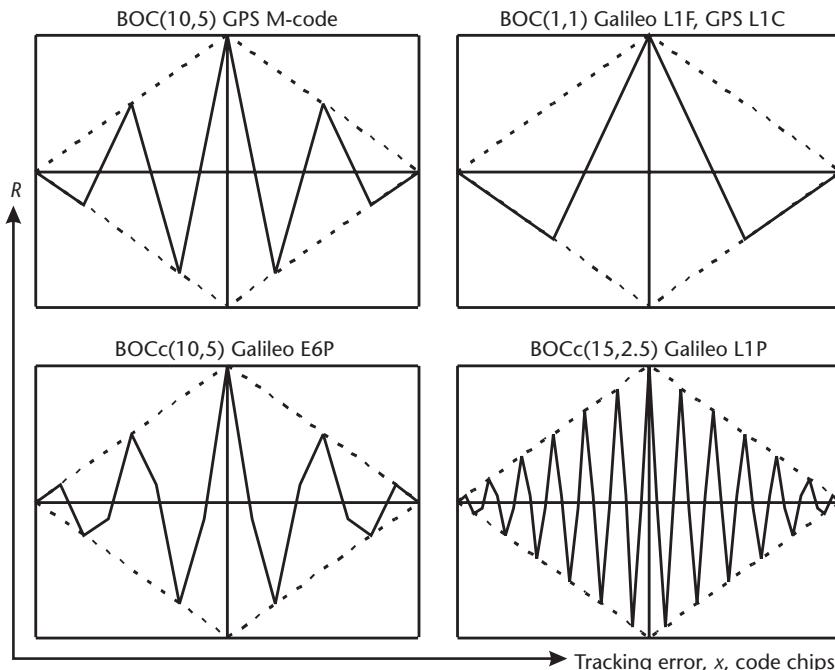


Figure 7.14 BOC combined spreading and subcarrier code correlation functions (neglecting band-limiting; dashed lines show equivalent BPSK correlation functions).

where E is the expectation operator. Averaging the noise samples, $w_{I0}(t)$, over each code chip to give samples, $w_{Ico,i}$, indexed by chip,

$$\sigma_{IQ}^2 = \frac{f_a^2}{f_{co}^2} \sum_{i=i_0}^{i_0 + f_{co}\tau_a - 1} \sum_{j=i_0}^{i_0 + f_{co}\tau_a - 1} E(w_{Ico,i} C_i w_{Ico,j} C_j) \quad (7.75)$$

noting that, in practice, the number of samples per chip is not an integer. As successive code chips are uncorrelated, $E(C_i C_j) = \delta_{ij}$, so

$$\sigma_{IQ}^2 = \frac{f_a^2}{f_{co}} \tau_a E(w_{Ico}^2) \quad (7.76)$$

If the noise is white with power spectral density n_0 , then the variance of the noise averaged over one code chip is $n_0 f_{co}$, provided that the precorrelation bandwidth, B_{PC} , is at least twice the chipping rate, f_{co} . From (7.63) and (7.59), the signal and noise amplitudes are boosted through the receiver front-end and AGC by a factor of $A_0/\sqrt{2P}$. Therefore

$$E(w_{Ico}^2) = \frac{A_0^2 n_0 f_{co}}{2P} \quad (7.77)$$

and

$$\sigma_{IQ} = A_0 f_a \sqrt{\frac{n_0 \tau_a}{2P}} \quad (7.78)$$

Where the noise is not white, the PSD should be weighted by the GNSS signal spectrum:

$$\bar{n}_0 = \frac{\int_{-\infty}^{\infty} n_0(f) s_a(f) df}{\int_{-\infty}^{\infty} s_a(f) df} \quad (7.79)$$

The ratio of the received signal power, P , to the weighted noise PSD, \bar{n}_0 , is known as the carrier power to noise density, c/n_0 . Thus, from, (7.74) and (7.78), the postcorrelation signal to noise amplitude ratio is

$$\frac{A_C}{\sigma_{IQ}} = \sqrt{2(c/n_0) \tau_a} \quad (7.80)$$

while the power ratio is

$$\frac{P_C}{\sigma_{IQ}^2} = \frac{\frac{1}{2} A_C^2}{\sigma_{IQ}^2} = (c/n_0) \tau_a \quad (7.81)$$

The carrier power to noise density is the primary measure of the signal-to-noise environment used to determine GNSS performance. It is commonly expressed in decibel form where, to avoid confusion, the upper case equivalent, C/N_0 , is used (some authors use C/No):

$$C/N_0 = 10 \log_{10}(c/n_0), \quad c/n_0 = 10^{\frac{C/N_0}{10}} \quad (7.82)$$

Determination of the carrier power to noise density as a function of the signal strength, interference levels, receiver, and antenna design is discussed in [3, 21]. For a strong GPS C/A-code signal at normal incidence to a good antenna in the absence of interference, C/N_0 may exceed 45 dB-Hz. Measurement of c/n_0 by the user equipment is discussed in Section 7.3.6, while the effect of c/n_0 on range measurement errors is discussed in Section 7.4.3.

7.2.4.3 Noise Properties

From (7.66), the noise standard deviation on the I and Q accumulated correlator outputs is σ_{IQ} by definition. This depends on four factors: the noise standard deviation prior to sampling, the quantization level applied at the ADC, the accumulation interval, and any scaling applied to the baseband processor's I and Q outputs. The AGC maintains a constant ratio between the quantization level and presampling noise standard deviation. Therefore, for all receivers with a continuous AGC, σ_{IQ} is constant for a given accumulation interval but varies between different receiver designs.

The normalized noise terms in the correlator outputs have unit variance by definition and zero mean. There is also no correlation between the noise in the in-phase and quadraphase channels because the product of I_c and Q_c averages close to zero over the accumulation interval. Thus, the noise terms have the following expectations

$$\begin{aligned} E(w_{I\alpha}^2) &= E(w_{Q\alpha}^2) = 1 \\ E(w_{I\alpha}) &= E(w_{Q\alpha}) = 0, \quad \alpha \in E, P, L \\ E(w_{I\alpha}w_{Q\beta}) &= 0 \end{aligned} \quad (7.83)$$

The noise on the early, prompt, and late correlator outputs is correlated because the same noise sequences, $w_{I0}(t_{sa})$ and $w_{Q0}(t_{sa})$, are multiplied by the same reference codes, offset by less than one chip. Thus, the noise is the same over the proportion of the correlation interval where the reference codes are aligned. Where precorrelation band-limiting is neglected, the correlation properties are

$$\begin{aligned} E(w_{IE}w_{IP}) &= E(w_{QE}w_{QP}) = E(w_{IP}w_{IL}) = E(w_{QP}w_{QL}) = 1 - d/2 \\ E(w_{IE}w_{IL}) &= E(w_{QE}w_{QL}) = 1 - d \end{aligned} \quad (7.84)$$

Band-limiting increases the correlation between the early, prompt, and late correlator outputs because it introduces time correlation to the input noise sequences, $w_{I0}(t_{sa})$ and $w_{Q0}(t_{sa})$ [22].

7.2.4.4 Accumulation Time

The choice of the time interval, τ_a , over which to accumulate the correlator outputs is a tradeoff between four factors: signal to noise, signal coherence, navigation-data-bit handling, and ranging-processor bandwidth. As (7.80) and (7.81) show, the signal-to-noise ratio of the baseband signal processor's I and Q outputs is optimized by maximizing τ_a . However, the other factors are optimized by short accumulation times.

If the residual carrier phase error after carrier wipeoff varies over the accumulation interval, the summed samples will interfere with each other. This is accounted for by the $\text{sinc}(\pi\delta f_{ca}\tau_a)$ term in (7.66). Total cancellation occurs where the phase error changes by an integer number of cycles over the accumulation interval. To maximize the I and Q signal to noise, a constant phase error must be maintained over the accumulation time. This is known as maintaining signal coherence, while linear summation of the Is and Qs is known as coherent integration. To limit the signal power loss due to carrier phase interference to a factor of 2, the following conditions must be met

$$\text{sinc}(\pi\delta f_{ca}\tau_a) < 1/\sqrt{2}$$

$$|\delta f_{ca}| < \frac{0.443}{\tau_a} \quad (7.85)$$

$$|\delta\dot{\phi}| < 0.443 \frac{c}{f_{ca}\tau_a}$$

Thus, for $\tau_a = 20$ ms and a signal in the L1 band, the range-rate error must be less than 4.2 m s^{-1} .

Squaring and adding the Is and Qs eliminates carrier phase interference over periods longer than the accumulation interval, so summation of $I^2 + Q^2$ is known as noncoherent integration. However, the power signal to noise for noncoherent integration varies as the square root of the integration time, as opposed to linearly for coherent integration. Therefore, signal to noise is optimized by performing coherent summation up to the point where carrier phase interference starts to be a problem and then performing noncoherent summation beyond that.

All of the legacy GNSS signals and about half of the new signals incorporate a navigation data message. If coherent summation is performed over two message data bits, there is an equal probability of those bits having the same or opposite signs. Where the bits are different, the signal component changes sign halfway through the summation, and the accumulated signal power is cancelled out. Accumulating correlator outputs over more than one data bit also prevents navigation message demodulation. Therefore, the data-bit length acts as the effective limit to the accumulation time for data-carrying signals, varying from 1 ms for the Galileo

E6C-d to 20 ms for the legacy signals, GPS L2C, and Galileo E5a-d (see Chapter 6). Specialist techniques for circumventing this limit are discussed in Section 8.3.6.

For the newer navigation-data-modulated GNSS signals, the code repetition interval is greater than or equal to the data bit length, so the data bit edges are determined from the ranging code. However, the GPS and GLONASS C/A codes repeat 20 times per data bit, requiring the data bit edges to be found initially using a search process (Section 7.3.5). Where the data bit edges are unknown, the only way of preventing summation across data bit boundaries is to limit the accumulation time to the 1-ms code length. However, for accumulation times below 20 ms, less than a quarter of the signal power is lost through summation across data bit boundaries.

The final factor in determining the accumulation time is the Nyquist criteria for the ranging processor. In tracking mode, the sampling rate of the tracking function must be at least twice the tracking loop bandwidth, which is larger for carrier tracking. The tracking function sampling rate is generally the inverse of the accumulation time, τ_a , and is known as the postcorrelation or predetection bandwidth. In acquisition mode, longer accumulation times result in either a longer acquisition time or the need to employ more correlators (see Section 7.3.1).

The optimum accumulation time depends on whether the ranging processor is in acquisition or tracking mode and can also depend on the signal-to-noise environment. The accumulation time can be varied within the baseband processor. However, it is simpler to fix it at the minimum required, typically 1 ms, and perform additional summation in the ranging processor.

7.2.4.5 Signal Multiplex Processing

The GPS L5, L1C, and most Galileo signals are broadcast as in-phase and quadrature pairs with a navigation data message on one component only. The GPS L2C and M-code signals are broadcast as time-division multiplexes with their codes alternating between navigation-message-modulated and unmodulated bits. Many receivers will process both signals, and many of the baseband processor functions can be shared between them. As the code and carrier of these multiplexed signal pairs are always in phase, the code and carrier NCOs can be shared.

A TDM signal pair may be treated as a single signal up until the code correlator outputs, with samples corresponding to alternate reference-code bits sent to separate data and pilot-channel accumulators. Alternatively, separate code correlators may be used for each component, with the reference code taking values of +1, -1, and 0, such that the reference code for one component is zero when the code for the other component is nonzero [23, 24].

Separate code correlators must be used for the in-phase and quadrature signal multiplexes, as both components are transmitted simultaneously. However, the in-phase and quadrature sampling phase may be shared, noting that the in-phase samples for one signal are the quadrature samples for the other.

7.3 Ranging Processor

The GNSS ranging processor uses the accumulated correlator outputs from the receiver to determine the pseudo-range, pseudo-range rate, and carrier phase and

to control the receiver's generation of the reference code and carrier. This section describes acquisition of GNSS signals and tracking of the code and carrier, followed by a discussion of tracking lock detection, navigation-message demodulation, signal-to-noise measurement, and generation of the measurements output to the navigation processor. Note that many processing techniques for BOC signals were still under development at the time of this writing.

7.3.1 Acquisition

When GNSS user equipment is switched on or a new satellite signal comes into view, the code phase of that signal is unknown. To determine this and obtain the time of signal transmission, the reference code phase must be varied until it matches that of the signal. Where one of the reference codes is within one chip of the signal code, the despread signal is observed in the receiver's accumulated correlator outputs (see Section 7.2.4). However, the Doppler-shifted carrier frequency of the signal must also be known to sufficient accuracy to maintain signal coherence over the accumulation interval. Otherwise, the reference Doppler shift must also be varied. This searching process is known as *acquisition* [1, 3, 14].

Each code phase and Doppler shift searched is known as a *bin*, while each combination of the two is known as a *cell*. The time spent correlating the signal is known as the *dwell time* and may comprise coherent and noncoherent integration. The code-phase bins are usually set half a chip apart. Except for long dwell times (see Section 8.3.4), the spacing of the Doppler bins is dictated by the coherent integration interval, τ_a , and is around $1/2\tau_a$. For each cell, a test statistic combining the in-phase and quadrature channels, $I^2 + Q^2$, is compared against a threshold. If the threshold is exceeded, the signal is deemed to be found.

Conventional acquisition algorithms, such as the Tong method, start at the center of the Doppler search window and move outward, alternating from side to side. Each code phase at a given Doppler shift is searched before moving onto the next Doppler bin. Code phase is searched from early to late so that directly received signals are usually found before reflected signals. As each baseband processor channel has three I and Q correlator pairs, three code phases may be searched simultaneously. Narrow correlator spacing should not be used for acquisition. Parallel baseband processor channels may be used to increase the number of parallel cells searched or for acquisition of other signals. When the acquisition threshold is exceeded, the test for that cell is repeated with further samples and the search is stopped if signal acquisition is confirmed.

Acquisition searches can cover more than one peak. Smaller auto-correlation and cross-correlation peaks arise in a code-phase search due to the limitations in the code correlation function (see Section 7.2.4.1); the longer the code repetition length, the smaller these peaks are. Smaller peaks arise in a Doppler search due to the minor peaks in the sinc function of (7.66), as illustrated by Figure 7.15. These are the same for all GNSS signals and are larger than the code-phase minor peaks. To prevent the acquisition algorithm from finding a minor peak first, the threshold may be set higher than the minor peaks in a strong signal-to-noise environment. The threshold is only reduced if no signal is found on the first search, noting that

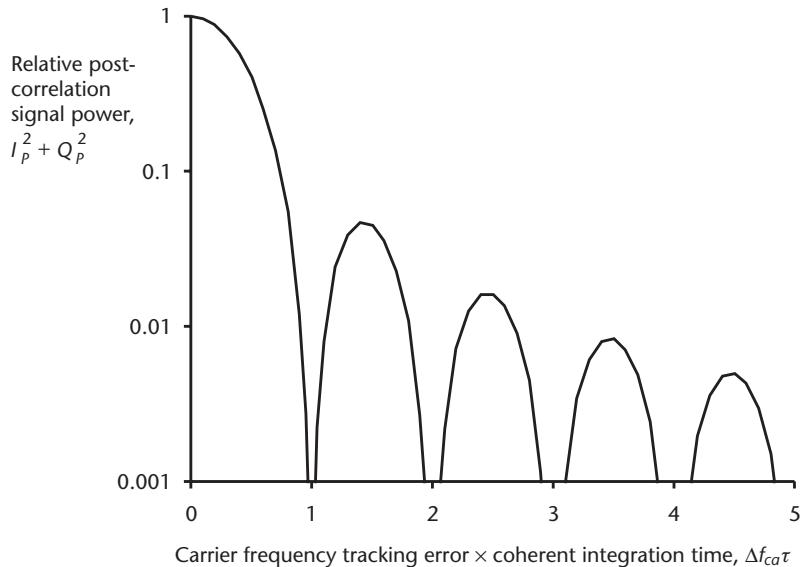


Figure 7.15 Relative postcorrelation signal power as a function of frequency tracking error.

the signal-to-noise level cannot be measured until a signal has been acquired; only the noise level can.

Where an AGC is used, the noise level is constant. Otherwise, the threshold is varied with the noise level to maintain a constant probability of false acquisition. The probability of missed detection thus increases as the signal to noise decreases. It may be reduced by increasing the dwell time. The greatest benefit is obtained by increasing the coherent integration time. However, this requires a reduced Doppler bin spacing, so the processing load increases with the square of the coherent integration time.

In most cases, the user equipment has the approximate time and the user position, almanac, and ephemeris data from when it was last used or through integration with other navigation systems. Situations where this information is not available prior to acquisition are known as *cold starts*. The size of the code search window is thus determined by the time and position uncertainty and can be set to the 3σ bounds. For short codes, such as the GPS and GLONASS C/A codes, the search window is limited to the code repetition length, significantly reducing the number of cells to search when prior information is poor. Conversely, very long codes, such as the GPS P(Y) and M codes and the Galileo PRS, cannot practically be acquired without prior knowledge of time. For a given search window, the number of code bins is directly proportional to the chipping rate. The size of the Doppler search window depends on the satellite and user velocity uncertainty and again is generally set to the 3σ bounds up to a maximum determined by the maximum user velocity and maximum satellite velocity along the line of sight ($\sim 900 \text{ m s}^{-1}$). However, where a low-cost reference oscillator (Section 7.2.2) is used, a wider Doppler search window may be needed for the first signal acquired in order to determine the receiver clock drift.

Where four or more satellite signals have been acquired, enabling calculation of a navigation solution, and current almanac data is available, the code search

window for acquiring signals from further satellites or reacquiring lost signals is small, while a Doppler search may not be required at all. The search window is similarly small where other signals from the same satellite have already been acquired. So, in most PPS GPS user equipment, the P(Y) code is acquired after the C/A code.

For a cold-start acquisition, the number of GPS C/A-code bins is 2046. In a strong signal-to-noise environment, an adequate dwell time is 1 ms, giving about 20 Doppler bins with a stationary receiver. Searching three cells at a time, acquisition can take place using a single channel within about 15 seconds. However, acquisition of low- C/N_0 signals using longer dwell times, acquiring higher chipping-rate codes, and acquiring codes with longer repetition lengths where prior information is poor all takes much longer using conventional techniques. Note that higher chipping-rate codes offer greater resistance against narrowband interference, while there is a requirement to acquire the GPS M code independently of other signals.

There are two solutions to the more challenging acquisition tasks. Receivers with massively parallel correlator arrays can search thousands of code bins in parallel, as opposed to 30–36 cells per frequency band for standard receivers [25, 26]. The fast Fourier transform (FFT) approach takes FFTs of the signal samples and the reference code at a given Doppler shift over the correlation accumulation interval, τ_a , and then performs the correlation in the frequency domain. An inverse FFT then produces test statistics for all of the code bins simultaneously [15, 27, 28].

The GPS L5 codes and most of the Galileo codes are layered. This can be used to speed up acquisition. By initially limiting the coherent integration interval to the primary-code repetition interval, the number of code bins in the acquisition search is limited to the primary-code length. Once the primary-code phase has been acquired, the full code is acquired. A longer coherent integration interval and dwell time must be used to capture the difference between secondary codes, but the number of code bins required is no greater than the secondary-code length.

Acquisition of a full BOC signal requires a code bin separation of a quarter of a subcarrier-function repetition interval because of the narrowing of the correlation function peak (see Section 7.2.4.1). However, if acquisition is performed using one side-lobe only or by combining the side-lobes noncoherently, the code bin separation need only be half a spreading-code chip. This reduces the number of cells to search by a factor of 2 for the Galileo L1F and GPS L1C signals, a factor of 4 for GPS M code and Galileo E6P, and a factor of 12 for Galileo L1P.

7.3.2 Code Tracking

Once a GNSS signal has been acquired, the code tracking process uses the I and Q measurements from the baseband signal processor to refine its measurement of the code phase, which is used to control the code NCO, maintaining the reference code's alignment with the signal. The code phase is also used to calculate the pseudo-range measurement as described in Section 7.3.7. Most GNSS user equipment performs code tracking for each signal independently using a fixed-gain delay lock loop (DLL) [18], while a Kalman filter may also be used [29]. Code tracking

may also be combined with navigation-solution determination as described in Section 7.5.3.

Figure 7.16 shows a typical code tracking loop. The early, prompt, and late in-phase and quadrature accumulated correlator outputs from the receiver are input to a discriminator function, which calculates a measurement of the code tracking error. This is used to correct the tracking loop's code-phase estimate, which is then predicted forward in time and used to generate a code NCO command, which is sent to the receiver. The prediction phase is usually aided with range-rate information from the carrier tracking function, the navigation processor, or an INS or dead reckoning system. Each step is now described.

The discriminator function may coherently integrate the Is and Qs over a navigation-message bit. However, there is no benefit in performing noncoherent integration, as the tracking loop does this inherently. The Is and Qs accumulated over a total time, τ_a , are then used to produce a discriminator function, D , which is proportional to the code tracking error, x . The most common discriminators are the dot-product power (DPP), early-minus-late power (ELP), and early-minus-late envelope (ELE) noncoherent discriminators [1, 14]:

$$\begin{aligned} D_{DPP} &= (I_E - I_L)I_P + (Q_E - Q_L)Q_P \\ D_{ELP} &= (I_E^2 + Q_E^2) - (I_L^2 + Q_L^2) \\ D_{ELE} &= \sqrt{I_E^2 + Q_E^2} - \sqrt{I_L^2 + Q_L^2} \end{aligned} \quad (7.86)$$

noting that only the dot-product discriminator uses the prompt correlator outputs.

A coherent discriminator is less noisy but requires carrier phase tracking to be maintained to keep the signal power in the in-phase channel. However, as carrier phase tracking is much less robust than code tracking, only noncoherent discrimina-

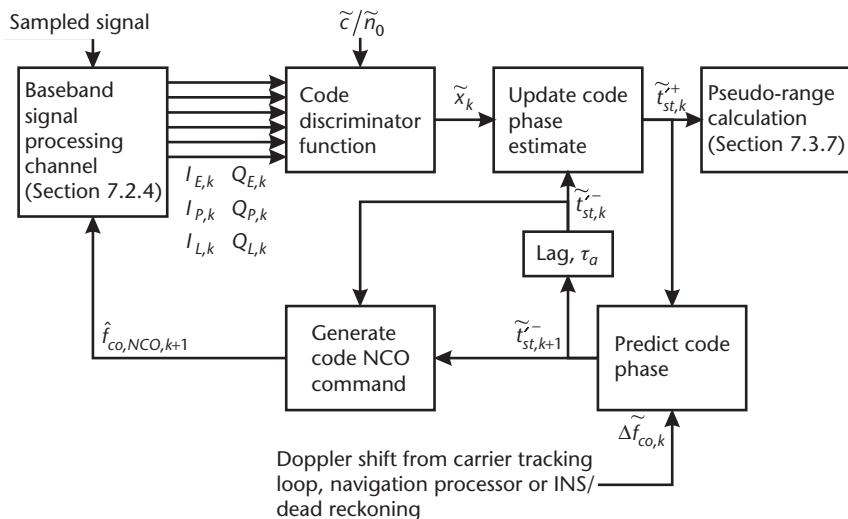


Figure 7.16 Code tracking loop.

tors may be used in poor signal-to-noise environments. An example of a coherent discriminator is

$$D_{Coh} = (I_E - I_L) \operatorname{sign}(I_P) \quad (7.87)$$

These discriminators all work on the principle that the signal power in the early and late correlation channels is equal when the prompt reference code is synchronized with the signal. In order to obtain a measurement of the code tracking error, the discriminator must be multiplied by a normalization function, N . Thus,

$$\tilde{x}_k = ND \quad (7.88)$$

where

$$N = \frac{\lim_{x \rightarrow 0} E[D(x)]}{x} \quad (7.89)$$

noting that the discriminator functions are only linear functions of x where x is small.

From (7.66), (7.70), (7.83), and (7.84), neglecting precorrelation band-limiting and assuming $|x| < 1 - d/2$ and $\delta f_{ca} \approx 0$, the expectations of the discriminator functions are

$$\begin{aligned} E(D_{DPP}) &\approx 2\sigma_{IQ}^2(c/n_0)\tau_a(1-|x|)(|x+d/2|-|x-d/2|) \\ E(D_{ELP}) &\approx 2\sigma_{IQ}^2(c/n_0)\tau_a(2-|x+d/2|-|x-d/2|)(|x+d/2|-|x-d/2|) \\ E(D_{ELE}) &\approx \sigma_{IQ}\sqrt{2(c/n_0)\tau_a}(|x+d/2|-|x-d/2|) \\ E(D_{Coh}) &\approx \sigma_{IQ}\sqrt{2(c/n_0)\tau_a}(|x+d/2|-|x-d/2|)\cos(\delta\phi_{ca}) \end{aligned} \quad (7.90)$$

From (7.89), the normalization functions are thus

$$\begin{aligned} N_{DPP} &= \frac{1}{4\sigma_{IQ}^2(\tilde{c}/\tilde{n}_0)\tau_a} \\ N_{ELP} &= \frac{1}{4(2-d)\sigma_{IQ}^2(\tilde{c}/\tilde{n}_0)\tau_a} \\ N_{ELE} &= \frac{1}{2\sigma_{IQ}\sqrt{2(\tilde{c}/\tilde{n}_0)\tau_a}} \\ N_{Coh} &= \frac{1}{2\sigma_{IQ}\sqrt{2(\tilde{c}/\tilde{n}_0)\tau_a}} \end{aligned} \quad (7.91)$$

noting that the measured carrier power to noise density has been substituted for its true counterpart. In some receivers, the normalization is performed by dividing

by $I^2 + Q^2$ or its root (as appropriate). However, $I^2 + Q^2$ is only proportional to $(c/n_0)\tau_a$ in strong signal-to-noise environments.

Figure 7.17 shows the discriminator input-output curves, neglecting noise, for early-late correlator spacings of 0.1 and 1 chips. With the larger correlator spacing, the discriminator can respond to larger tracking errors. It has a larger pull-in range and a longer linear region. However, as shown in Section 7.4.3, the tracking noise is larger.

Where a pair of signals from the same satellite in the same frequency band, one with and one without a navigation message (see Section 7.2.4.5), are both tracked, they may share a common tracking function, with the discriminator outputs from the two signals averaged. Where a longer coherent integration time is used for the pilot signal, its discriminator should be given higher weighting [23]. Alternatively, measurements from the pilot signal may be used to maintain tracking of both signals.

For tracking of BOC signals, the correlator spacing must be carefully selected so that the early, prompt, and late correlators all lie on the central peak of the correlation function (see Figure 7.14) to ensure that the discriminator function has the correct sign for small code tracking errors [30]. The discriminator function will still exhibit sign errors for large tracking errors [31], limiting the tracking errors that can be recovered from or locking onto a minor peak of the correlation function. The range of the discriminator can be extended by using additional very early and very late correlators to identify the side-lobes of the correlation function [32]. Alternatively, BOC_s(1,1) signals may be correlated with the equivalent prompt BPSK code in place of the early and late correlation channels, giving a cross-correlation function that may be used as the code discriminator [33].

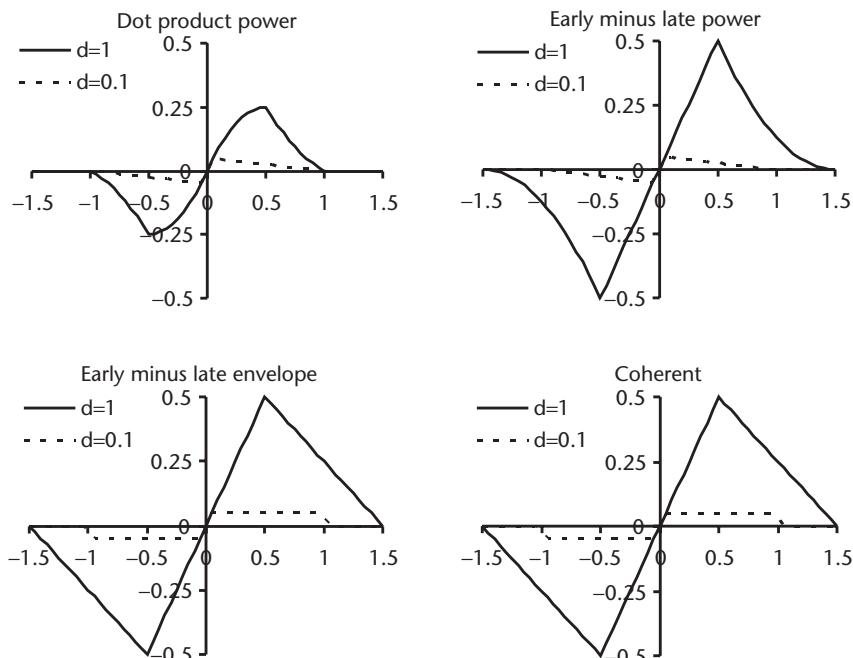


Figure 7.17 Code discriminator input-output curves (units: chips).

The code-phase estimate of the tracking function is denoted here by \tilde{t}'_{st} , as it is offset from the time of signal transmission, t_{st} , by an integer number of code repetition periods. This is updated using the discriminator output:

$$\tilde{t}'^+_{st,k} = \tilde{t}'^-_{st,k} + K_{co} \tilde{x}_k / f_{co} \quad (7.92)$$

where K_{co} is the code loop gain and, in analogy with Kalman filter notation, the subscript k denotes the iteration, while the superscripts – and + denote before and after the update, respectively. The loop gain is set at less than unity to smooth out the noise on the correlator outputs. The double-sided noise bandwidth of the code tracking loop, B_{L_CO} , is given by [1, 14]

$$B_{L_CO} = K_{co} / 4\tau_a \quad (7.93)$$

Conversely,

$$K_{co} = 4B_{L_CO} \tau_a \quad (7.94)$$

The code tracking bandwidth typically takes values between 0.05 and 1 Hz. The narrower the bandwidth, the greater the noise resistance, but the longer it takes to respond to dynamics (see Section 7.4.3). Thus, its selection is a tradeoff. The code tracking loop does not need to track the absolute code phase, only the error in the code phase obtained by integrating the range-rate aiding. The most accurate aiding is from a carrier phase tracking loop, in which case, the code tracking loop need only track the code-carrier divergence due to ionospheric dispersion (Section 7.4.2). However, carrier phase cannot always be tracked. Where a carrier frequency tracking loop or another navigation system provides the aiding, the code tracking loop must track the error in the range-rate aiding. Where both aiding sources are available, they may be weighted according to their respective uncertainties.

Where the aiding is provided by the GNSS navigation processor, it may only be able to supply the range rate due to the satellite motion and Earth rotation, leaving the code tracking loop to track the user dynamics. This is only possible where the user velocity is less than $4B_{L_CO}$ times the maximum recoverable tracking error (see Section 7.4.3). With a 1-Hz code tracking bandwidth, land vehicle dynamics can be tracked using most GNSS signals.

Another issue is the navigation-solution update rate required. The interval between statistically independent code-phase measurements is approximately $1/4B_{L_CO}$. Consequently, the lowest code tracking bandwidths tend to be used for static applications.

The code-phase estimate is predicted forward to the next iteration using

$$\tilde{t}'^-_{st,k+1} = \tilde{t}'^+_{st,k} + \frac{f_{co} + \Delta\tilde{f}_{co,k}}{f_{co}} \tau_a \quad (7.95)$$

where f_{co} is the transmitted code chipping rate and $\Delta\tilde{f}_{co,k}$ is its Doppler shift, obtained from the aiding source. This may be calculated from the carrier Doppler shift or pseudo-range rate using

$$\Delta \tilde{f}_{co} = \frac{f_{co}}{f_{ca}} \Delta \tilde{f}_{ca} \approx -\frac{f_{co}}{c} \tilde{\rho}_R \quad (7.96)$$

Most GNSS receivers do not allow step changes to the reference code, so code-phase corrections are made by running the code NCO faster or slower than the Doppler-shifted code chipping rate. This can be done by setting the code NCO frequency to the following:

$$\hat{f}_{co, \text{NCO}, k+1} = \frac{\tilde{t}_{st, k+1}^+ - \tilde{t}_{st, k}^-}{\tau_a} f_{co} \quad (7.97)$$

Except in a software receiver, the processing of the code tracking function and the signal correlation occur simultaneously. Therefore, there is a lag of one correlation period, τ_a , in applying the NCO control corrections to the receiver. Figure 7.18 illustrates this. However, this is not a problem as the lag is much less than the time constant of the tracking loop.

7.3.3 Carrier Tracking

The primary purpose of carrier tracking in GNSS user equipment is to maintain a measurement of the Doppler-shifted carrier frequency. This is used to maintain signal coherence over the correlator accumulation interval. It is also used to aid the code tracking loop and to provide a less noisy measurement of the pseudo-range rate to the navigation processor. Either the carrier phase or the carrier frequency may be tracked, noting that a carrier phase tracking function also tracks the frequency.

Carrier phase tracking enables the navigation data message to be demodulated more easily and allows precision carrier-phase positioning techniques (Section 8.2) to be used. Carrier frequency tracking is more robust in poor signal-to-noise and high-dynamics environments. Consequently, many GNSS user equipment designs implement frequency tracking as a reversionary mode to phase tracking and as an intermediate step between acquisition and phase tracking.

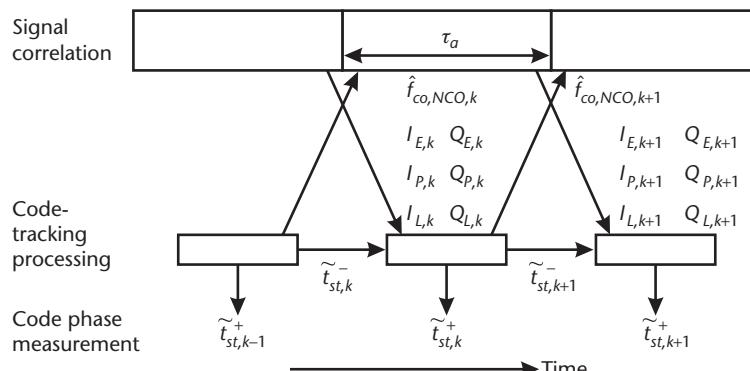


Figure 7.18 Timing of signal correlation and code tracking processing.

Most GNSS user equipment performs carrier tracking independently for each signal, using a fixed-gain phase lock loop (PLL) for phase tracking and a frequency lock loop (FLL) for frequency tracking. Figures 7.19 and 7.20 show typical carrier phase- and carrier frequency-tracking loops. These are similar to the code tracking loop. The main differences are that only the prompt correlator outputs from the baseband signal processor are used; there is usually no external aiding information; and the loop estimates three quantities for phase tracking and two for frequency tracking. The carrier frequency estimate aids maintenance of the carrier phase estimate and the rate-of-frequency-change estimate aids maintenance of the frequency estimate. A combined PLL and FLL using both types of discriminator may also be implemented [34]. Carrier tracking of BOC signals is no different from that of BPSK signals.

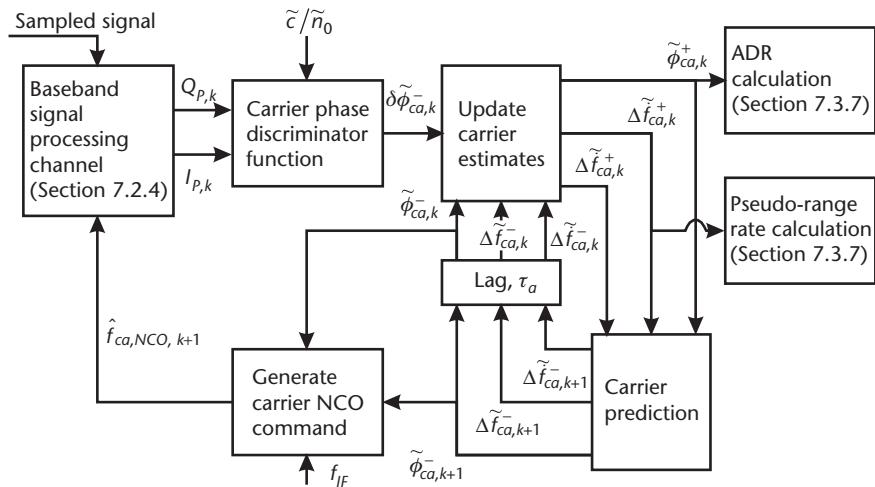


Figure 7.19 Carrier phase-tracking loop.

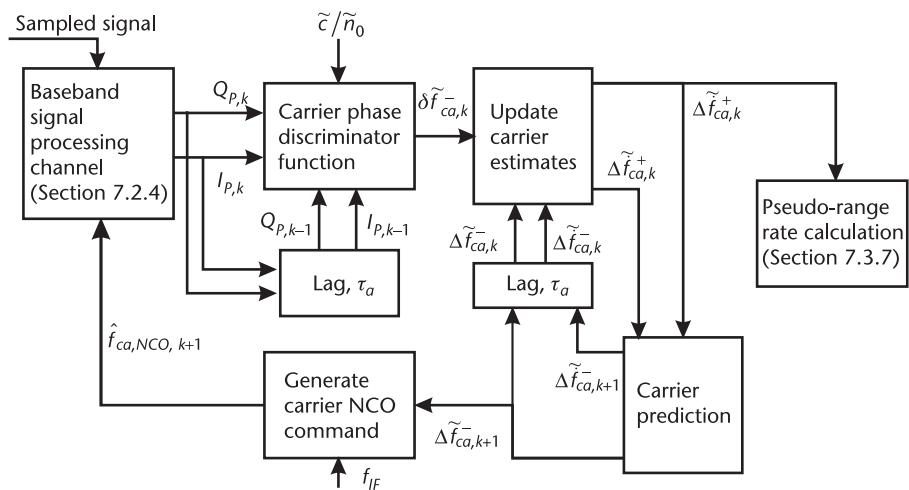


Figure 7.20 Carrier frequency-tracking loop.

For navigation message-modulated signals, the I and Q samples have a common-mode sign ambiguity due to the navigation data bit. To prevent this from disrupting carrier tracking, Costas discriminators may be used, which give the same result regardless of the data-bit sign. Examples include the IQ-product (IQP), decision-directed-Q (DDQ), Q-over-I (QOI), and two-quadrant arctangent (ATAN) discriminators [1, 14]:

$$\begin{aligned}\Phi_{IQP} &= Q_P I_P & \Phi_{DDQ} &= Q_P \operatorname{sign}(I_P) \\ \Phi_{QOI} &= Q_P / I_P & \Phi_{ATAN} &= \arctan(Q_P / I_P)\end{aligned}\quad (7.98)$$

The Costas discriminators are only sensitive to carrier phase tracking errors in the range $-90^\circ < \delta\phi_{ca} < 90^\circ$, exhibiting a sign error outside this range, while the Q-over-I discriminator exhibits singularities at $\pm 90^\circ$.

Costas discriminators may also be used for the pilot signals. However, it is better to use PLL discriminators, which are sensitive to the full range of tracking errors. Examples include the quadraphase-channel (QC) and four-quadrant arc-tangent (ATAN2) discriminators:

$$\Phi_{QC} = Q_P \quad \Phi_{ATAN2} = \arctan2(Q_P, I_P) \quad (7.99)$$

Where a common carrier tracking function is used for a pair of signals, with and without a navigation message, in the same frequency band, it is better to use the pilot signal for the carrier discriminator, though a weighted average may also be used [23].

To obtain a measurement of the carrier-phase error, the discriminator is normalized:

$$\tilde{\delta\phi}_{ca} = N\Phi \quad (7.100)$$

where

$$N = \frac{\lim_{\delta\phi \rightarrow 0} E[\Phi(\delta\phi_{ca})]}{\delta\phi_{ca}} \quad (7.101)$$

giving normalization functions of

$$\begin{aligned}N_{IQP} &= \frac{1}{2(\tilde{c}/\tilde{n}_0) \tau_a} \\ N_{DDQ} &= N_{QC} = \frac{1}{\sqrt{2(\tilde{c}/\tilde{n}_0) \tau_a}} \\ N_{QOI} &= N_{ATAN} = N_{ATAN2} = 1\end{aligned}\quad (7.102)$$

Figure 7.21 shows the discriminator input-output curves. Note that the Costas discriminator functions repeat every 180° , so a carrier-tracking loop using one is equally likely to track 180° out-of-phase as in-phase.

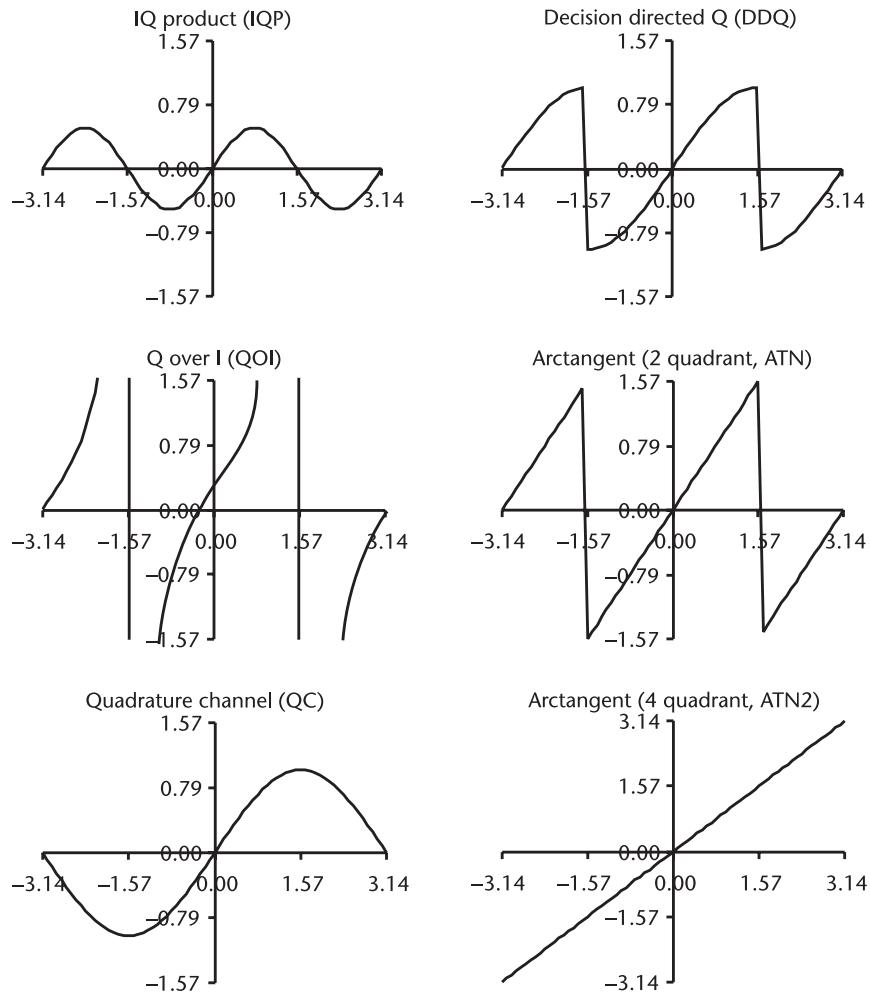


Figure 7.21 Carrier phase discriminator input-output curves (units: rad).

Carrier frequency discriminators use the current and previous correlator outputs. The decision-directed cross-product (DDC), cross-over-dot product (COD), and the ATAN discriminators are Costas discriminators and may be used across data-bit transitions:

$$\begin{aligned}
 F_{DDC} &= (I_{P,k-1}Q_{P,k} - I_{P,k}Q_{P,k-1}) \operatorname{sign}(I_{P,k-1}I_{P,k} + Q_{P,k-1}Q_{P,k}) \\
 F_{COD} &= \frac{I_{P,k-1}Q_{P,k} - I_{P,k}Q_{P,k-1}}{I_{P,k-1}I_{P,k} + Q_{P,k-1}Q_{P,k}} \\
 F_{ATAN} &= \arctan\left(\frac{I_{P,k-1}Q_{P,k} - I_{P,k}Q_{P,k-1}}{I_{P,k-1}I_{P,k} + Q_{P,k-1}Q_{P,k}}\right) \\
 &= \arctan\left(\frac{Q_{P,k}}{I_{P,k}}\right) - \arctan\left(\frac{Q_{P,k-1}}{I_{P,k-1}}\right)
 \end{aligned} \tag{7.103}$$

The cross-product (CP) and ATAN2 discriminators are FLL discriminators and cannot be used across data-bit transitions:

$$\begin{aligned} F_{CP} &= I_{P,k-1}Q_{P,k} - I_{P,k}Q_{P,k-1} \\ F_{ATAN2} &= \arctan2(I_{P,k-1}Q_{P,k} - I_{P,k}Q_{P,k-1}, I_{P,k-1}I_{P,k} + Q_{P,k-1}Q_{P,k}) \\ &= \arctan2(Q_{P,k}, I_{P,k}) - \arctan2(Q_{P,k-1}, I_{P,k-1}) \end{aligned} \quad (7.104)$$

To obtain a measurement of the carrier frequency error, the discriminator is normalized:

$$\delta\tilde{f}_{ca} = NF \quad (7.105)$$

where

$$N = \frac{\lim_{\delta\phi \rightarrow 0} E[F(\delta\tilde{f}_{ca})]}{\delta\tilde{f}_{ca}} \quad (7.106)$$

giving normalization functions of

$$N_{DDC} = N_{CP} = \frac{1}{4\pi(\tilde{c}/\tilde{n}_0)\tau_a^2} \quad (7.107)$$

$$N_{COD} = N_{ATAN} = N_{ATAN2} = \frac{1}{2\pi\tau_a}$$

where it is assumed that τ_a is the interval between I and Q samples as well as the accumulation period. Figure 7.22 shows the discriminator input-output curves. Note that the maximum frequency error for all discriminators is inversely proportional to the accumulation interval [14].

In a carrier phase tracking function, the PLL is typically third order and the estimates of the carrier phase, $\tilde{\phi}_{ca}$, Doppler frequency shift, $\Delta\tilde{f}_{ca}$, and rate of change of Doppler, $\Delta\tilde{f}_{ca}$, are updated using

$$\begin{aligned} \tilde{\phi}_{ca,k}^+ &= \tilde{\phi}_{ca,k}^- + K_{ca1}\delta\tilde{\phi}_{ca,k} \\ \Delta\tilde{f}_{ca,k}^+ &= \Delta\tilde{f}_{ca,k}^- + \frac{K_{ca2}}{2\pi\tau_a}\delta\tilde{\phi}_{ca,k} \\ \Delta\tilde{f}_{ca,k}^+ &= \Delta\tilde{f}_{ca,k}^- + \frac{K_{ca3}}{2\pi\tau_a^2}\delta\tilde{\phi}_{ca,k} \end{aligned} \quad (7.108)$$

where K_{ca1} , K_{ca2} , and K_{ca3} are the tracking loop gains and k , $-$, and $+$ are as defined for code tracking. The carrier phase tracking bandwidth is then [14]

$$B_{L_CA} = \frac{K_{ca1}^2 K_{ca2} + K_{ca2}^2 - K_{ca1} K_{ca3}}{4(K_{ca1} K_{ca2} - K_{ca3})\tau_a} \quad (7.109)$$

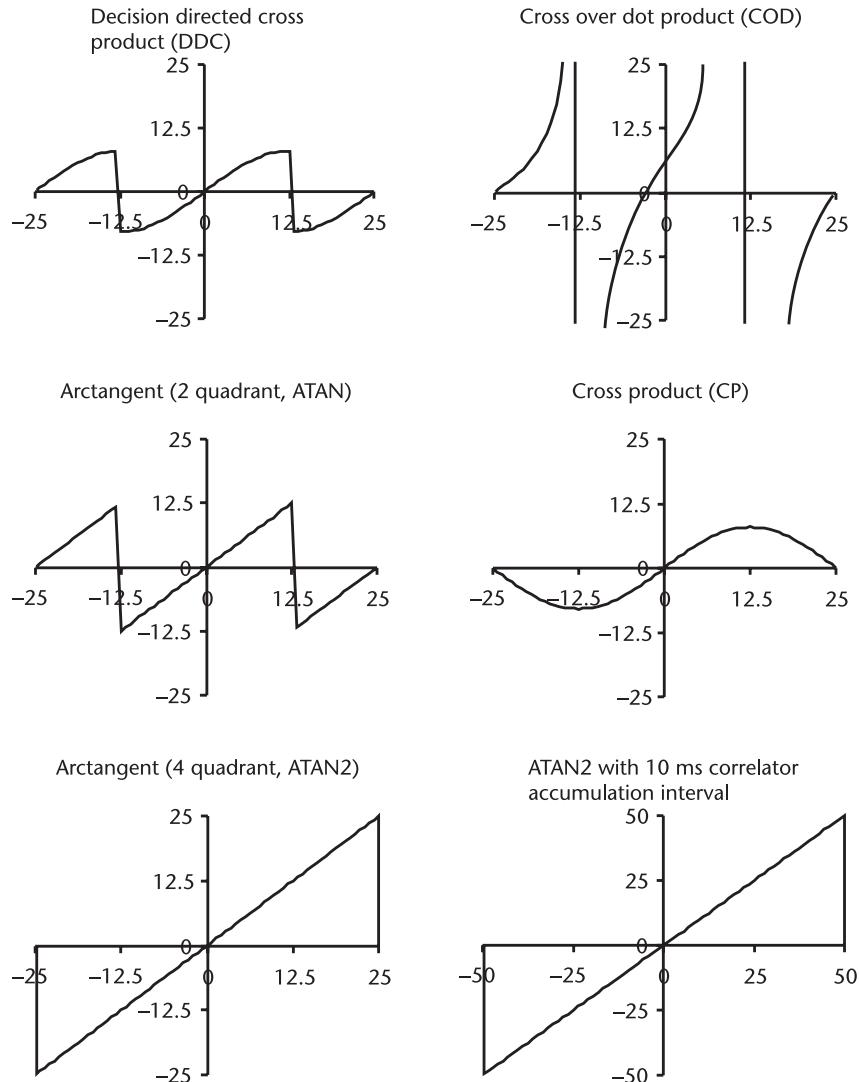


Figure 7.22 Carrier frequency discriminator input-output curves (units: Hz; $\tau_a = 20$ ms correlator accumulation interval, except where indicated otherwise).

A commonly used set of gains is [1]

$$K_{ca1} = 2.4B_{L_CA}\tau_a, \quad K_{ca2} = 2.88(B_{L_CA}\tau_a)^2, \quad K_{ca3} = 1.728(B_{L_CA}\tau_a)^3 \quad (7.110)$$

As for code tracking, narrower bandwidths give more noise smoothing, while wider bandwidths give better dynamics response. For applications where the user antenna is stationary, the signal dynamics change very slowly. However, the receiver's oscillator noise must be tracked in order to maintain carrier phase lock. A tracking bandwidth of 5 Hz is sufficient for this. Where the antenna is subject to high dynamics or vibration, a higher bandwidth of 15–18 Hz is needed in order to track changes in line-of-sight acceleration, or jerk.

The update phase of a carrier frequency-tracking function using a second-order FLL is

$$\begin{aligned}\Delta\tilde{f}_{ca,k}^+ &= \Delta\tilde{f}_{ca,k}^- + K_{cf1} \delta\tilde{f}_{ca,k} \\ \Delta\tilde{f}_{ca,k}^+ &= \Delta\tilde{f}_{ca,k}^- + \frac{K_{cf2}}{\tau_a} \delta\tilde{f}_{ca,k}\end{aligned}\quad (7.111)$$

and the carrier frequency tracking bandwidth is

$$B_{L-CF} = \frac{K_{cf1}^2 + K_{cf2}}{4K_{cf1}\tau_a} \quad (7.112)$$

with a typical value of 2 Hz [14].

The carrier phase tracking loop's estimates are predicted forward to the next iteration using

$$\begin{aligned}\tilde{\phi}_{ca,k+1}^- &= \tilde{\phi}_{ca,k}^+ + 2\pi(f_{IF} + \Delta\tilde{f}_{ca,k}^+) \tau_a + \pi\Delta\tilde{f}_{ca,k}^+ \tau_a^2 \\ \Delta\tilde{f}_{ca,k+1}^- &= \Delta\tilde{f}_{ca,k}^+ + \Delta\tilde{f}_{ca,k}^+ \tau_a \\ \Delta\tilde{f}_{ca,k+1}^- &= \Delta\tilde{f}_{ca,k}^+\end{aligned}\quad (7.113)$$

while the estimates of the frequency-tracking loop are predicted forward using

$$\begin{aligned}\Delta\tilde{f}_{ca,k+1}^- &= \Delta\tilde{f}_{ca,k}^+ + \Delta\tilde{f}_{ca,k}^+ \tau_a \\ \Delta\tilde{f}_{ca,k+1}^- &= \Delta\tilde{f}_{ca,k}^+\end{aligned}\quad (7.114)$$

The reference signal carrier phase in the receiver is advanced and retarded by running the carrier NCO faster or slower than the Doppler-shifted carrier frequency. Thus, in user equipment implementing carrier phase tracking, the carrier NCO frequency is set to

$$\hat{f}_{ca,NCO,k+1} = f_{IF} + \Delta\tilde{f}_{ca,k+1}^- + \frac{\tilde{\phi}_{ca,k}^+ - \tilde{\phi}_{ca,k}^-}{2\pi\tau_a} \quad (7.115)$$

Where carrier frequency tracking is used, the carrier NCO frequency is simply set to the ranging processor's best estimate:

$$\hat{f}_{ca,NCO,k+1} = f_{IF} + \Delta\tilde{f}_{ca,k+1}^- \quad (7.116)$$

Block-diagram treatments of the code and carrier loop filters may be found in other texts [1, 14].

7.3.4 Tracking Lock Detection

GNSS user equipment must detect when it is no longer tracking the code from a given signal so that contamination of the navigation processor with incorrect

pseudo-range data is avoided and the acquisition mode reinstigated to try and recover the signal.

Code can no longer be tracked when the tracking error exceeds the pull-in range of the discriminator. This is the region within which the discriminator output in the absence of noise has the same sign as the tracking error. As Figure 7.17 shows, the pull-in range depends on the correlator spacing and discriminator type.

Tracking lock is lost when the carrier power-to-noise density, C/N_0 , is too low and/or the signal dynamics is too high. Whether there is sufficient signal to noise to maintain code tracking is determined by measuring C/N_0 and comparing it with a minimum value. The threshold should match the code discriminator pull-in range to about three times the code tracking noise standard deviation (see Section 7.4.3) and allow a margin for C/N_0 measurement error. A threshold of around 19 dB-Hz is suitable with a 1-Hz code tracking bandwidth. The same test can be used to detect loss of code lock due to dynamics as this causes the measured C/N_0 to be underestimated [19].

Loss of carrier phase tracking lock must be detected to enable the ranging processor to transition to carrier frequency tracking and prevent erroneous Doppler measurements from disrupting code tracking and the navigation processor. As with code tracking, a C/N_0 measurement threshold can be used to determine whether there is sufficient signal to noise to track carrier phase. A suitable threshold is around 25 dB-Hz with an arctangent Costas discriminator. This will not detect dynamics-induced loss of lock. However, because the carrier discriminator function repeats every 180° (Costas) or 360° (PLL), carrier phase lock can be spontaneously recovered. During the interval between loss and recovery, the carrier phase estimate can advance or retard by a multiple of 180° or 360° with respect to truth. This is known as a cycle slip and affects the accumulated delta range measurements (see Section 7.3.7) and navigation-message demodulation (see Section 7.3.5). For applications where cycle-slip detection is required, a phase lock detector should be employed [1, 14].

Carrier frequency lock is essential for maintaining code tracking, as it ensures signal coherence over the correlator accumulation interval (see Section 7.2.4.4), while the C/N_0 level needed to maintain carrier frequency tracking is similar to that needed for code tracking. However, as Figure 7.22 shows, carrier frequency discriminators repeat a number of times within the main peak of the signal power versus tracking error curve (Figure 7.15). Consequently, the FLL can undergo false lock at an offset of $n/2\tau_a$ from the true carrier frequency, where n is an integer (assuming a Costas discriminator). This produces a pseudo-range-rate measurement error of a few m s^{-1} and disrupts navigation-message demodulation. A PLL can also exhibit false frequency lock following a cycle slip. To prevent this, a false-frequency-lock detector must be implemented. This simply compares the Doppler shift from the FLL with that obtained from the code tracking loop.

7.3.5 Navigation-Message Demodulation

In stand-alone GNSS user equipment, the navigation data message must be demodulated in order to obtain the satellite positions and velocities and resolve any ambiguities.

ties in the time of transmission. Where carrier phase tracking is in lock, the data bit is given simply by

$$D(t) = \text{sign}(I_P(t)) \quad (7.117)$$

Where carrier frequency tracking is used, the data-bit transitions are detected by observing the 180° changes in $\arctan 2(I_P, Q_P)$. This gives noisier data demodulation than phase tracking. In both cases, there is a sign ambiguity in the demodulated data-bit stream. In frequency tracking, this occurs because the sign of the initial bit is unknown, whereas in phase tracking, it occurs because it is unknown whether the tracking loop is locked in-phase or 180° out-of-phase. The ambiguity is resolved using the parity check information broadcast in the message itself. This must be checked continuously, as phase tracking is vulnerable to cycle slips and frequency tracking to missed detection of the bit transitions [1].

The signal to noise on the data demodulation is optimized by matching the correlator accumulation interval (Section 7.2.4.4) to the length of the data bit. Accumulating over data-bit transitions should be avoided. For the newer GNSS signals, the timing of the bit transitions is indicated by the ranging code. However, for GPS and GLONASS C/A code, there is an ambiguity, as there are 20 code repetition intervals per data bit, so the ranging processor has to search for the bit transitions. This is done by forming 20 test statistics, each summed coherently over 20 ms with a different offset and then noncoherently over n data bits. The test statistics are

$$T_r = \sum_{i=1}^n \left(\sum_{j=1}^{20} I_{P,(20i+j+r)} \right)^2 \quad (7.118)$$

for carrier phase tracking and

$$T_r = \sum_{i=1}^n \left[\left(\sum_{j=1}^{20} I_{P,(20i+j+r)} \right)^2 + \left(\sum_{j=1}^{20} Q_{P,(20i+j+r)} \right)^2 \right] \quad (7.119)$$

for frequency tracking, where r takes values from 1 to 20 and the accumulation interval for the Is and Qs is 1 ms. The largest test statistic corresponds to the correct bit synchronization [14, 29].

7.3.6 Carrier-Power-to-Noise-Density Measurement

Measurements of the carrier power to noise density, c/n_0 , (see Section 7.2.4.2) are needed for tracking lock detection. They may also be used to determine the weighting of measurements in the navigation processor and for adapting the tracking loops to the signal-to-noise environment.

To correctly determine receiver performance, c/n_0 must be measured after the signal is correlated with the reference code. Precorrelation signal-to-noise measure-

ment techniques overestimate c/n_0 where the interference is concentrated near the center of the signal spectrum and underestimate it where the interference is concentrated away from the main signal peak [35, 36].

The narrow-to-wide power-ratio measurement method computes the coherently summed narrowband power, P_N , and noncoherently summed wideband power, P_W , over an interval τ_{aN} , generally the data-bit interval:

$$P_N = \left(\sum_{i=1}^M I_{P,i} \right)^2 + \left(\sum_{i=1}^M Q_{P,i} \right)^2 \quad P_W = \sum_{i=1}^M (I_{P,i}^2 + Q_{P,i}^2) \quad (7.120)$$

where $I_{P,i}$ and $Q_{P,i}$ are accumulated over time $\tau_{aW} = \tau_{aN}/M$, typically 1 ms. The power ratio is then computed and averaged over n iterations to reduce noise:

$$\bar{P}_{N/W} = \frac{1}{n} \sum_{r=1}^n \frac{P_{N,r}}{P_{W,r}} \quad (7.121)$$

Taking expectations [1, 19],

$$E(\bar{P}_{N/W}) \approx \frac{M[(c/n_0)\tau_{aN} + 1]}{M + (c/n_0)\tau_{aN}} \quad (7.122)$$

The carrier-power-to-noise-density measurement is then

$$\tilde{c}/\tilde{n}_0 = \frac{M(\bar{P}_{N/W} - 1)}{\tau_{aN}(M - \bar{P}_{N/W})} \quad (7.123)$$

The correlator-comparison method compares the narrow-band power with the output, I_N , of a noise correlation channel, which correlates with an unused PRN code. The measurement averaged over n iterations is [19]

$$Z_{cc} = \frac{\sum_{k=1}^n (I_{P,k}^2 + Q_{P,k}^2)}{\sum_{k=1}^n I_{N,k}^2} \quad (7.124)$$

where $I_{P,k}$, $Q_{P,k}$, and $I_{N,k}$ are accumulated over time τ_a . Taking expectations,

$$E(Z_{cc}) \approx [(c/n_0)\tau_a + 1] \left(1 + \frac{2}{n} \right) \quad (7.125)$$

giving

$$\tilde{c}/\tilde{n}_0 = \frac{1}{\tau_a} \left(\frac{n}{n+2} Z_{cc} - 1 \right) \quad (7.126)$$

C/N_0 is obtained using (7.82). A discriminator-statistics method may also be used [19]. All of these methods are very noisy at low c/n_0 , requiring a long averaging time to produce useful measurements. The averaging time may be varied to optimize the tradeoff between noise and response time.

7.3.7 Pseudo-Range, Pseudo-Range-Rate, and Carrier-Phase Measurements

GNSS ranging processors output three types of measurement: pseudo-range, pseudo-range rate or Doppler shift, and accumulated delta range (ADR). The pseudo-range measurement is obtained from code tracking and the others from carrier tracking.

The pseudo-range is given by

$$\tilde{\rho}_R = (\tilde{t}_{sa} - \tilde{t}_{st})c \quad (7.127)$$

where \tilde{t}_{sa} is the time of signal arrival, measured by the receiver clock, and \tilde{t}_{st} is the measured time of signal transmission. To obtain the transmission time from the code phase, \tilde{t}'_{st} , measured by the code tracking loop (Section 7.3.2), an integer number of code repetition periods, determined from the navigation message, must usually be added.

For the GPS and GLONASS C/A codes, the additional step of determining the data-bit transitions must also be performed (see Section 7.3.5). Bit-synchronization errors produce errors in \tilde{t}_{st} of multiples of 1 ms, leading to pseudo-range errors of multiples of 300 km. The navigation processor should check for these errors.

The Doppler-shift measurement, $\Delta\tilde{f}_{ca}$, is obtained directly from the carrier tracking loop (Section 7.3.3). This may be transformed to a pseudo-range-rate measurement using

$$\tilde{\rho}_R = \frac{-c\Delta\tilde{f}_{ca}(2f_{ca} + \Delta\tilde{f}_{ca})}{2f_{ca}^2 + 2f_{ca}\Delta\tilde{f}_{ca} + \Delta\tilde{f}_{ca}^2} = -\frac{c\Delta\tilde{f}_{ca}}{f_{ca}} \left(1 - \frac{\Delta\tilde{f}_{ca}}{2f_{ca}} + \dots \right) \approx -\frac{c}{f_{ca}} \Delta\tilde{f}_{ca} \quad (7.128)$$

The ADR, $\Delta\tilde{\rho}_{ADR}$, is the integral of the pseudo-range rate:

$$\Delta\tilde{\rho}_{ADR}(t_{sa}) = \int_{t_0}^{t_{sa}} \tilde{\rho}_R(t) dt \quad (7.129)$$

where t_0 is the time of carrier tracking initialization. In practice, it is obtained from the carrier NCO. The delta range is similar, but is only integrated over the interval since the last measurement. The navigation processor will only use one carrier-derived measurement as they all convey the same information. The ADR and delta-range measurements have the advantage of smoothing out the carrier tracking noise where the navigation-processor iteration rate is less than the carrier-tracking bandwidth.

The duty cycles of the tracking loops are commonly aligned with the navigation-data-bit transitions and/or the code repetition period. Consequently, the tracking loops for the different signals are not synchronized, producing measurements corresponding to different times of arrival. However, navigation-solution computation is much simpler if a common time of signal arrival can be assumed. Consequently, the measurements are typically predicted forward to a common time of arrival as described in [14].

The code and carrier measurements can be combined to produce a smoothed pseudo-range, $\tilde{\rho}_S$:

$$\tilde{\rho}_S(t) = W_{co}\tilde{\rho}_R(t) + (1 - W_{co})[\tilde{\rho}_S(t - \tau) + \Delta\tilde{\rho}_{ADR}(t) - \Delta\tilde{\rho}_{ADR}(t - \tau)] \quad (7.130)$$

where W_{co} is the code weighting factor and τ the update interval. This improves the accuracy of a single-point position solution or receiver autonomous integrity monitoring (RAIM) algorithm (see Section 15.4.1), but does not benefit filtered positioning or integrity monitoring, where smoothing is implicit. The time constant, τ/W_{co} , is typically set at 100s, limiting the effects of code–carrier ionosphere divergence (see Section 7.4.2) and cycle slips. This smoothes the code tracking noise by about an order of magnitude [37].

GNSS user equipment may output raw pseudo-ranges and pseudo-range rates or it may apply corrections for the satellite clock, ionosphere propagation, and troposphere propagation errors as described in the next section. Some user equipment also subtracts the Sagnac correction, $\delta\rho_{ie}$, forcing the navigation processor to use the ECEF coordinate frame; this is not always documented.

7.4 Range Error Sources

The pseudo-range, pseudo-range rate, and ADR measurements made by GNSS user equipment are subject to two main types of error: time-correlated and noise-like. The satellite clock errors and the ionosphere and troposphere propagation errors are correlated over the order of an hour and are partially corrected for by the user equipment using (7.43). The errors prior to the application of corrections are known as raw errors and those remaining after the correction process are known as residual errors.

Tracking errors are correlated over less than a second and cannot be corrected, only smoothed. Errors due to multipath are typically correlated over a few seconds and can be mitigated using a number of techniques, as discussed in Section 8.4.

This section describes each of these error sources in turn, together with the ephemeris prediction error, which affects the navigation solution through the computation of the satellite position and velocity.

Note that the receiver clock offset and drift are treated as unknown terms in the navigation solution, rather than as error sources.

7.4.1 Satellite Clock and Ephemeris Prediction Errors

The satellite clock error arises due to the cumulative effect of oscillator noise. It is mostly corrected for using three calibration coefficients, a_{f0} , a_{f1} , and a_{f2} , and

a reference time, t_{oc} , transmitted in the navigation data message. In addition, a relativistic correction is applied to account for the variation in satellite clock speed with the velocity and gravitational potential over the satellite's elliptical orbit [9, 38]. The total satellite clock correction for satellite j is

$$\Delta\rho_{scj} = [a_{f0,j} + a_{f1,j}(t_{st,j} - t_{oc}) + a_{f2,j}(t_{st,j} - t_{oc})^2]c - 2\frac{\mathbf{r}_{esj}^e \cdot \mathbf{v}_{esj}^e}{c} \quad (7.131)$$

where a $\pm 604,800$ seconds correction is applied to t_{oc} where $|t_{st,j} - t_{oc}| > 302,400$ seconds to account for week crossovers.

The residual clock error depends on the size of the control segment's monitor network, the latency of the corrections, and the stability of the satellite clock itself. The average residual clock error for the GPS constellation in 2005, following the L-AII, was 1.0m [39]. However, as the rubidium clocks installed in the Block-IIR and later satellites are about an order of magnitude more stable over the data latency period than the cesium clocks used by the older satellites, this value will drop as the constellation is updated [2, 40].

The ephemeris prediction error is simply the error in the control segment's prediction of the satellite position. Its components, expressed in orbital-frame cylindrical coordinates, δr_{osj}^o , δu_{osj}^o , and δz_{osj}^o , as shown in Figure 7.23, are correlated over the order of an hour and change each time the ephemeris data in the navigation message is updated. The range error due to the ephemeris error is

$$\delta\rho_{ej} = \frac{\mathbf{r}_{\beta sj}^\beta}{|\mathbf{r}_{\beta sj}^\beta|} \cdot \mathbf{u}_{asj}^\beta \delta r_{osj}^o + \frac{\mathbf{v}_{\beta sj}^\beta}{|\mathbf{v}_{\beta sj}^\beta|} \cdot \mathbf{u}_{asj}^\beta r_{osj}^o \delta u_{osj}^o + \frac{\mathbf{r}_{\beta sj}^\beta \wedge \mathbf{v}_{\beta sj}^\beta}{|\mathbf{r}_{\beta sj}^\beta| |\mathbf{v}_{\beta sj}^\beta|} \cdot \mathbf{u}_{asj}^\beta \delta z_{osj}^o, \quad \beta \in i, e, I \quad (7.132)$$

This varies with the signal geometry, so is different for users at different locations, but is dominated by the radial component, δr_{osj}^o . The ephemeris errors are reduced by employing a larger monitor station network, as this enables better

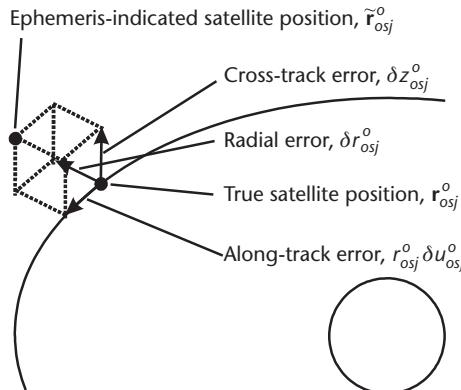


Figure 7.23 Components of the ephemeris prediction error.

separation of the three ephemeris error components and the satellite clock error. The average GPS ephemeris range error, following the L-AII in 2005, was 0.45m [39].

7.4.2 Ionosphere and Troposphere Propagation Errors

GNSS signals are refracted by free electrons in the ionosphere, which extends from about 50 to 1,000 km above the Earth's surface, and by gases in the troposphere, which extends to about 12 km above the surface. Signals from low-elevation satellites experience much more refraction than signals from high-elevation satellites as they pass through more atmosphere, as Figure 7.24 shows.

The ionosphere propagation delay varies with the elevation angle approximately as [3, 9]

$$\delta\rho_{ij} \propto \left[1 - \left(\frac{R \cos \theta_{mu,j}}{R + h_i} \right)^2 \right]^{-1/2} \quad (7.133)$$

where R is the average Earth radius and h_i is the mean ionosphere height, about 350 km. The troposphere propagation delay varies approximately as [3, 9]

$$\delta\rho_{tj} \propto \left[1 - \left(\frac{\cos \theta_{mu,j}}{1.001} \right)^2 \right]^{-1/2} \quad (7.134)$$

These are known as obliquity factors or mapping functions and are shown in Figure 7.25. They are unity for satellites at zenith or normal incidence, which is 90° elevation. Most GNSS user equipment implements a minimum elevation threshold, known as the mask angle, of between 5° and 10°, below which signals are excluded from the navigation solution.

The ionosphere is a dispersive medium, meaning that the propagation velocity varies with the frequency. As with nondispersive refraction, the signal modulation

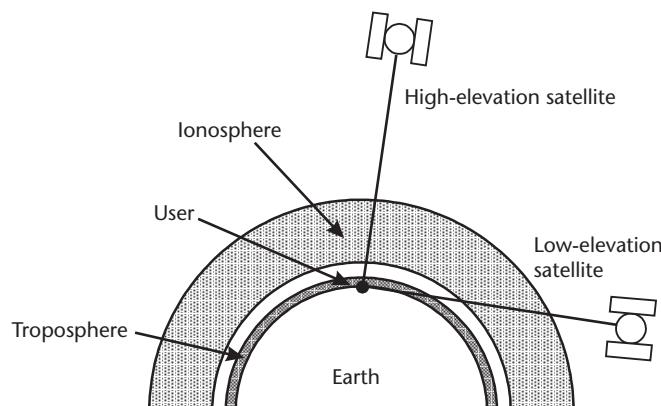


Figure 7.24 Ionosphere and troposphere propagation for high- and low-elevation satellites (not to scale).

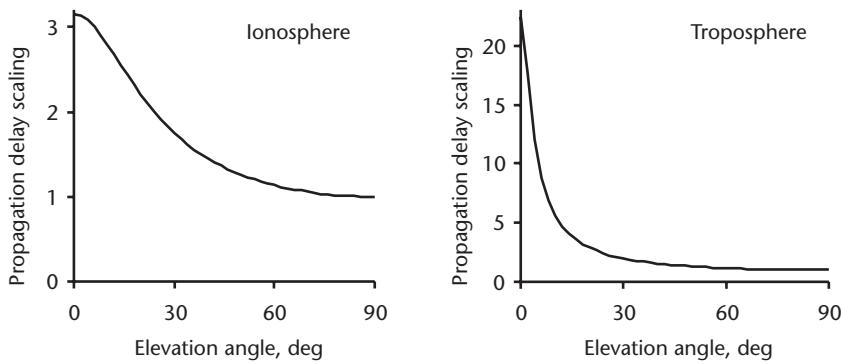


Figure 7.25 Ionosphere and troposphere delay mapping functions.

(PRN code and navigation data) is delayed. However, the carrier phase is advanced by the same amount, as explained in [9]. Consequently, code- and carrier-based range measurements gradually diverge. More than 99 percent of the propagation delay/advance varies as f_{ca}^{-2} [41]. Ionization of the ionosphere gases is caused by solar radiation, so there is more refraction during the day than at night. The signal modulation delay for a satellite at zenith varies from 1–3m around 02:00 to 5–15m around 14:00 local time.

The troposphere is a nondispersive medium, so all GNSS signals are delayed equally and there is no code-carrier divergence. The total delay at zenith is about 2.5m. About 90 percent of the delay is attributable to the dry gases in the atmosphere and is relatively stable. The remaining 10 percent is due to water vapor, so varies with the weather.

Where a GNSS receiver tracks signals on more than one frequency, they may be combined to eliminate most of the ionosphere propagation delay. The ionosphere-corrected pseudo-range is then

$$\tilde{p}_{Ricj} = \frac{f_{ca,a}^2 \tilde{p}_{Ra,j} - f_{ca,b}^2 \tilde{p}_{Rb,j}}{f_{ca,a}^2 - f_{ca,b}^2} \quad (7.135)$$

where the subscripts a and b denote the two frequencies. However, this brings a penalty in the form of increased tracking noise. The tracking error standard deviation of the corrected pseudo-range is [42]

$$\sigma_{ptic} = \frac{\sqrt{f_{ca,a}^4 \sigma_{pta}^2 + f_{ca,b}^4 \sigma_{ptb}^2}}{|f_{ca,a}^2 - f_{ca,b}^2|} \quad (7.136)$$

where σ_{pta} and σ_{ptb} are the code tracking error standard deviations on the two frequencies. The closer together the two frequencies, the more the tracking error is scaled up. For users of GPS L1 and L2 signals, $\sigma_{ptic}/\sigma_{ptL1} \approx 3.36$, noting that $\sigma_{ptL2}/\sigma_{ptL1} \approx \sqrt{2}$ due to the different transmission powers.

As the ionosphere propagation delay varies slowly, with correlation times of about half an hour [40], the ionosphere correction may be smoothed over time to

reduce the tracking error. Applying the smoothing over m iterations and using k to denote the current iteration, the corrected pseudo-range is then

$$\tilde{\rho}_{Ricj,k} = \tilde{\rho}_{Raj,k} + \Delta\rho_{Ricj,k} \quad (7.137)$$

where

$$\Delta\rho_{Ricj,k} = \frac{f_{ca,b}^2}{m(f_{ca,a}^2 - f_{ca,b}^2)} \sum_{i=k+1-m}^k (\tilde{\rho}_{Raj,i} - \tilde{\rho}_{Rbj,i}) \quad (7.138)$$

Alternatively, if the measurements on the two frequencies are weighted to minimize the tracking noise,

$$\tilde{\rho}_{Ricj,k} = W_a \tilde{\rho}_{Raj,k} + (1 - W_a) \tilde{\rho}_{Rbj,k} + \Delta\rho_{Ricj,k} \quad (7.139)$$

where

$$\Delta\rho_{Ricj,k} = \frac{(1 - W_a)f_{ca,a}^2 + W_a f_{ca,b}^2}{m(f_{ca,a}^2 - f_{ca,b}^2)} \sum_{i=k+1-m}^k (\tilde{\rho}_{Raj,i} - \tilde{\rho}_{Rbj,i}) \quad (7.140)$$

and

$$W_a = \frac{\sigma_{\rho tb}^2}{\sigma_{\rho ta}^2 + \sigma_{\rho tb}^2} \quad (7.141)$$

The residual ionosphere propagation error following smoothed dual-frequency correction is of the order of 0.1m [9].

The carrier-smoothed pseudo-range may be corrected for the ionosphere propagation delay using

$$\begin{aligned} \tilde{\rho}_{Sicj}(t) &= \frac{W_{co}(f_{ca,a}^2 \tilde{\rho}_{Raj}(t) - f_{ca,b}^2 \tilde{\rho}_{Rbj}(t))}{f_{ca,a}^2 - f_{ca,b}^2} \\ &\quad + (1 - W_{co}) \left[\tilde{\rho}_{Sicj}(t - \tau) + \frac{\left[\begin{array}{l} f_{ca,a}^2(\Delta\tilde{\rho}_{ADRaj}(t) - \Delta\tilde{\rho}_{ADRaj}(t - \tau)) \\ f_{ca,b}^2(\Delta\tilde{\rho}_{ADRbj}(t) - \Delta\tilde{\rho}_{ADRbj}(t - \tau)) \end{array} \right]}{f_{ca,a}^2 - f_{ca,b}^2} \right] \end{aligned} \quad (7.142)$$

As this also corrects for code-carrier ionosphere divergence, it allows a longer smoothing time constant to be used [37].

Single-frequency users use a model to estimate the ionosphere propagation delay as a function of time, the user latitude and longitude, the elevation and azimuth of each satellite line of sight, and parameters broadcast in the navigation data message. For GPS, the Klobuchar model is used, which incorporates eight

broadcast parameters, common to all satellites [5, 41], while Galileo uses the NeQuick model, incorporating three broadcast parameters [43]. These models correct about 50 percent of the propagation delay. GLONASS does not broadcast any ionosphere data.

Models are also used to correct the troposphere propagation error as a function of elevation angle and user height. A number of different models are in use as described in [3, 4, 44, 45]. Residual errors are of the order of 0.2m [9], but may be reduced by incorporating temperature, pressure, and humidity measurements. Incorporation of meteorological sensors is not practical for most GNSS user equipment. One option is to transmit weather forecast data to users requiring high precision [46, 47].

During periods of high solar storm activity, the ionosphere refractive index can fluctuate on a localized basis from second to second, a process known as scintillation. This has two effects: the carrier phase of the received signal can fluctuate rapidly, and interference between signals taking different paths to the user can cause rapid fading. Furthermore, smoothing of the dual-frequency ionosphere correction becomes invalid. Scintillation is most common in equatorial regions between sunset and midnight, but can also occur in polar regions [21].

7.4.3 Tracking Errors

The code and carrier discriminator functions, described in Sections 7.3.2 and 7.3.3, exhibit random errors due to receiver thermal noise, RF interference, and other GNSS signals on the same frequency. Neglecting precorrelation band-limiting, it may be shown [48] that the code discriminator noise variances for a BPSK signal are

$$\begin{aligned}\sigma^2(N_{DPP} D_{DPP}) &\approx \frac{d}{4(c/n_0) \tau_a} \left[1 + \frac{1}{(c/n_0) \tau_a} \right] \\ \sigma^2(N_{ELP} D_{ELP}) &\approx \frac{d}{4(c/n_0) \tau_a} \left[1 + \frac{2}{(2-d)(c/n_0) \tau_a} \right] \\ \sigma^2(N_{ELE} D_{ELE}) &\approx \frac{d}{4(c/n_0) \tau_a}, (c/n_0) \tau_a \gg 1 \\ \sigma^2(N_{Coh} D_{Coh}) &\approx \frac{d}{4(c/n_0) \tau_a}\end{aligned}\quad (7.143)$$

where the infinite precorrelation bandwidth approximation is valid for $d \geq \pi f_{co}/B_{PC}$. The variances of D_{DPP} and D_{ELP} under the trapezium approximation are given in [19], while [22] discusses the general case.

The tracking loop smoothes out the discriminator noise, but also introduces a lag in responding to dynamics. The code discriminator output may be written as

$$\tilde{x}_k = k_{ND}(x_k)x_k + w_{ND} \quad (7.144)$$

where w_{ND} is the discriminator noise and k_{ND} is the discriminator gain, which may be obtained from the slopes of Figure 7.17. The gain is unity for small tracking

errors by definition, but drops as the tracking error approaches the pull-in limits of the discriminator. From (7.67), (7.92), and (7.144), the code tracking error is propagated as

$$\begin{aligned} x_k^+ &= x_k^- - K_{co}(k_{ND}x_k^- + w_{ND}) \\ &= (1 - K_{co}k_{ND})x_k^- - K_{co}w_{ND} \end{aligned} \quad (7.145)$$

The code tracking error has zero mean and standard deviation σ_x , while the discriminator noise has zero mean and standard deviation σ_{ND} as given by (7.143). Squaring (7.145) and applying the expectation operator,

$$\sigma_x^2 = (1 - K_{co}\bar{k}_{ND})^2\sigma_x^2 + K_{co}^2\sigma_{ND}^2 \quad (7.146)$$

Assuming $K_{co} \ll 1$ and $\bar{k}_{ND} \approx 1$, which is valid except on the verge of tracking loss,

$$\sigma_x^2 \approx \frac{1}{2}K_{co}\sigma_{ND}^2 \quad (7.147)$$

Substituting in (7.94),

$$\sigma_x^2 \approx 2B_{L_CO}\tau_a\sigma_{ND}^2 \quad (7.148)$$

The code tracking noise standard deviation in chips, neglecting precorrelation band-limiting, is thus

$$\begin{aligned} \sigma_x &\approx \sqrt{\frac{B_{L_CO}d}{2(c/n_0)} \left[1 + \frac{1}{(c/n_0)\tau_a} \right]} \quad D = D_{DPP} \\ &\sqrt{\frac{B_{L_CO}d}{2(c/n_0)} \left[1 + \frac{2}{(2-d)(c/n_0)\tau_a} \right]} \quad D = D_{ELP} \\ &\sqrt{\frac{B_{L_CO}d}{2(c/n_0)}} \quad D = D_{ELE}, (c/n_0)\tau_a \gg 1 \text{ or } D = D_{Coh} \end{aligned} \quad (7.149)$$

The standard deviation with an early-minus-late power discriminator, accounting for precorrelation band-limiting, is given in [14, 22]. The pseudo-range error standard deviation due to tracking noise is

$$\sigma_{ptj} = \frac{c}{f_{co}}\sigma_{xj} \quad (7.150)$$

where j denotes the satellite or tracking channel.

For a BOC(f_s, f_{co}) signal where precorrelation band-limiting can be neglected, the code tracking noise standard deviation is that of a f_{co} chipping-rate BPSK signal

multiplied by $\frac{1}{2}\sqrt{f_{co}/f_s}$ [49]. In practice, this only applies to a BOC_s(1,1) signal as the other GNSS BOC signals require narrow correlator spacings, so the effect of precorrelation band-limiting is significant. The tracking noise standard deviation for the GPS M code is given in [14, 30].

The carrier phase discriminator noise variance is [1, 3]

$$\sigma^2(N\Phi) \approx \frac{1}{2(c/n_0)\tau_a} \left[1 + \frac{1}{2(c/n_0)\tau_a} \right] \quad (7.151)$$

with a Costas discriminator and

$$\sigma^2(N\Phi) \approx \frac{1}{2(c/n_0)\tau_a} \quad (7.152)$$

with a PLL discriminator. The carrier tracking noise standard deviation is then

$$\begin{aligned} \sigma_{\delta\phi} &\approx \sqrt{\frac{B_{L_CA}}{(c/n_0)} \left[1 + \frac{1}{2(c/n_0)\tau_a} \right]} && \text{Costas} \\ &\approx \sqrt{\frac{B_{L_CA}}{(c/n_0)}} && \text{PLL} \end{aligned} \quad (7.153)$$

Without cycle slips, the ADR standard deviation due to tracking noise is

$$\sigma(\Delta\tilde{\rho}_{ADRj}) = \frac{c}{2\pi f_{ca}} \sigma_{\delta\phi} \quad (7.154)$$

The carrier frequency tracking noise standard deviation with carrier phase tracking is

$$\sigma_{\delta f} \approx \sqrt{\frac{0.72B_{L_CA}}{\tau_a} \frac{\sigma_{\delta\phi}}{2\pi}} \quad (7.155)$$

while, with carrier frequency tracking, it is [14]

$$\sigma_{\delta f} \approx \frac{1}{2\pi\tau_a} \sqrt{\frac{4B_{L_CF}}{(c/n_0)} \left[1 + \frac{1}{(c/n_0)\tau_a} \right]} \quad (7.156)$$

The pseudo-range-rate error standard deviation due to tracking noise is

$$\sigma_{rt} = \frac{c}{f_{ca}} \sigma_{\delta f} \quad (7.157)$$

The code tracking error due to the lag in responding to dynamics depends on the tracking loop bandwidth and error in the range-rate aiding, $\tilde{\rho}_R - \dot{\rho}_R$. The steady-state tracking error due to a constant range-rate error is

$$\begin{aligned}\delta\rho_{w_lag,j} &= \frac{(\tilde{\rho}_R - \dot{\rho}_R)\tau_a}{K_{co}} = \frac{\tilde{\rho}_R - \dot{\rho}_R}{4B_{L_CA}} \\ x_{lag} &= \frac{f_{co}}{4B_{L_CA}c} (\tilde{\rho}_R - \dot{\rho}_R)\end{aligned}\quad (7.158)$$

Note that with a 1-Hz code tracking bandwidth, a 20-ms coherent integration interval, and a BPSK signal, range-rate aiding errors will cause loss of signal coherence before the code tracking error is pushed outside the pull-in range of the code discriminator.

A third-order carrier phase tracking loop does not exhibit tracking errors in response to velocity or acceleration, but is susceptible to line-of-sight jerk. The steady state ADR and phase tracking errors due to a constant line-of-sight jerk are [1]

$$\delta\Delta\rho_{ADR_lag,j} = -\frac{\ddot{\rho}_{Rj}}{(1.2B_{L_CA})^3}, \quad \delta\phi_{ca_lag,j} = \frac{2\pi f_{ca}}{c} \frac{\ddot{\rho}_{Rj}}{(1.2B_{L_CA})^3} \quad (7.159)$$

where the tracking loop gains in (7.110) are assumed. To prevent cycle slips, the jerk must thus be limited to

$$|\ddot{\rho}_R| < \frac{(1.2B_{L_CA})^3 c}{4f_{ca}} \quad (7.160)$$

with a Costas discriminator and twice this with a PLL discriminator, noting that the threshold applies to the average jerk over the time constant of the carrier tracking loop, $1/4B_{L_CA}$. In practice, the threshold should be set lower to prevent cycle slips due to a mixture of jerk and noise. The steady-state range-rate and Doppler errors are

$$\delta\dot{\rho}_{w_lag,j} = -\frac{\ddot{\rho}_{Rj}}{(1.2B_{L_CA})^2}, \quad \delta f_{ca_lag,j} = \frac{f_{ca}}{c} \frac{\ddot{\rho}_{Rj}}{(1.2B_{L_CA})^2} \quad (7.161)$$

A second-order carrier frequency tracking loop exhibits the following steady-state errors due to a constant jerk [1, 14]

$$\delta\dot{\rho}_{w_lag,j} = -\frac{\ddot{\rho}_{Rj}}{(1.885B_{L_CF})^2}, \quad \delta f_{ca_lag,j} = \frac{f_{ca}}{c} \frac{\ddot{\rho}_{Rj}}{(1.885B_{L_CF})^2} \quad (7.162)$$

To prevent false lock, the line-of-sight jerk must then be limited to

$$|\ddot{\rho}_R| < \frac{(1.885B_{L_CF})^2 c}{2f_{ca} \tau_a} \quad (7.163)$$

with a Costas discriminator and twice this otherwise. Again, a lower threshold should be set in practice due to noise.

7.4.4 Multipath

Multipath interference can occur where the user equipment receives reflected signals from a given satellite in addition to the direct signals. For land applications, signals are generally reflected off the ground, buildings, or trees, as shown in Figure 7.26, while for aircraft and ships, reflections off the host-vehicle body are more common. Interference can also occur from diffracted signals. The reflected and diffracted signals are always delayed with respect to the direct signals and have a lower amplitude unless the direct signals are attenuated (e.g., by a building or foliage). Low-elevation-angle signals are usually subject to the greatest multipath interference.

Each reflected or diffracted signal may be described by an amplitude, α_i , range lag, Δ_i , and carrier phase offset, $\delta\phi_{mi}$, with respect to the direct signal. There is also a carrier frequency offset, δf_{mi} , which is larger where the user is moving with respect to the reflecting surface [50]. By analogy with (7.59), the total received signal is then

$$s_a(t_{sa}) = \sqrt{2P} \sum_{i=0}^n \left\{ \alpha_i C(t_{st} - \Delta_i/c) D(t_{st} - \Delta_i/c) \right. \\ \left. \times \cos[2\pi(f_{ca} + \Delta f_{ca} + \delta f_{mi})t_{sa} + \phi_{ca} + \delta\phi_{mi}] \right\} \quad (7.164)$$

where n is the number of reflected or diffracted signals and $\alpha_0 = 1$, while $\Delta_0 = \delta\phi_{m0} = \delta f_{m0} = 0$. The accumulated correlator outputs, given by (7.66), then become

$$\begin{aligned} I_E(t_{sa}) &= \sigma_{IQ} \left\{ \sqrt{2(c/n_0)\tau_a} D(t_{st}) \sum_{i=0}^n \left[\alpha_i R(x - \delta_i - d/2) \operatorname{sinc}(\pi(\delta f_{ca} + \delta f_{mi})\tau_a) \right] + w_{IE}(t_{sa}) \right\} \\ I_P(t_{sa}) &= \sigma_{IQ} \left\{ \sqrt{2(c/n_0)\tau_a} D(t_{st}) \sum_{i=0}^n \left[\alpha_i R(x - \delta_i) \operatorname{sinc}(\pi(\delta f_{ca} + \delta f_{mi})\tau_a) \right] + w_{IP}(t_{sa}) \right\} \\ I_L(t_{sa}) &= \sigma_{IQ} \left\{ \sqrt{2(c/n_0)\tau_a} D(t_{st}) \sum_{i=0}^n \left[\alpha_i R(x - \delta_i + d/2) \operatorname{sinc}(\pi(\delta f_{ca} + \delta f_{mi})\tau_a) \right] + w_{IL}(t_{sa}) \right\} \\ Q_E(t_{sa}) &= \sigma_{IQ} \left\{ \sqrt{2(c/n_0)\tau_a} D(t_{st}) \sum_{i=0}^n \left[\alpha_i R(x - \delta_i - d/2) \operatorname{sinc}(\pi(\delta f_{ca} + \delta f_{mi})\tau_a) \right] + w_{QE}(t_{sa}) \right\} \\ Q_P(t_{sa}) &= \sigma_{IQ} \left\{ \sqrt{2(c/n_0)\tau_a} D(t_{st}) \sum_{i=0}^n \left[\alpha_i R(x - \delta_i) \operatorname{sinc}(\pi(\delta f_{ca} + \delta f_{mi})\tau_a) \right] + w_{QP}(t_{sa}) \right\} \\ Q_L(t_{sa}) &= \sigma_{IQ} \left\{ \sqrt{2(c/n_0)\tau_a} D(t_{st}) \sum_{i=0}^n \left[\alpha_i R(x - \delta_i + d/2) \operatorname{sinc}(\pi(\delta f_{ca} + \delta f_{mi})\tau_a) \right] + w_{QL}(t_{sa}) \right\} \end{aligned} \quad (7.165)$$

where the effect of multipath on navigation data reception is neglected and $\delta_i = \Delta_i f_{co}/c$ is the lag in code chips.

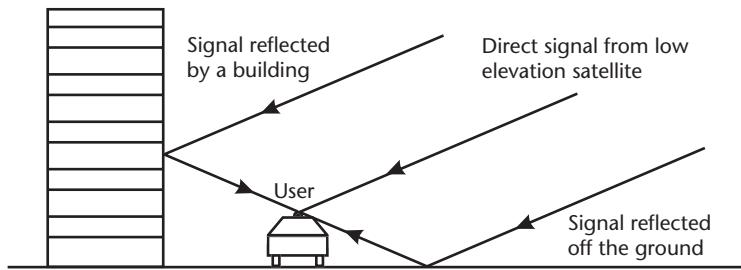


Figure 7.26 Example of a multipath interference scenario.

Tracking errors due to multipath are maximized when the carrier phase offset is 0° and 180° . The multipath interference is constructive where $-90^\circ < \delta\phi_{mi} < 90^\circ$ and destructive otherwise. Figure 7.27 shows the direct signal, reflected signal, and combined correlation functions for a single interfering signal with $\delta = 1/4$, $\alpha = 1/2$, and $\delta\phi_{mi} = 0$ and 180° ; precorrelation band-limiting is neglected. Note that there is no interference from the main correlation peak where $\delta > 1 + d/2$. Consequently, higher chipping-rate signals are less susceptible to multipath interference as the range lag, Δ , must be smaller for the reflected signal to affect the main correlation peak.

The code tracking loop acts to equate the signal powers in the early and late correlation channels, so the tracking error in the presence of multipath is obtained by solving

$$I_E^2 + Q_E^2 - I_L^2 - Q_L^2 = 0 \quad (7.166)$$

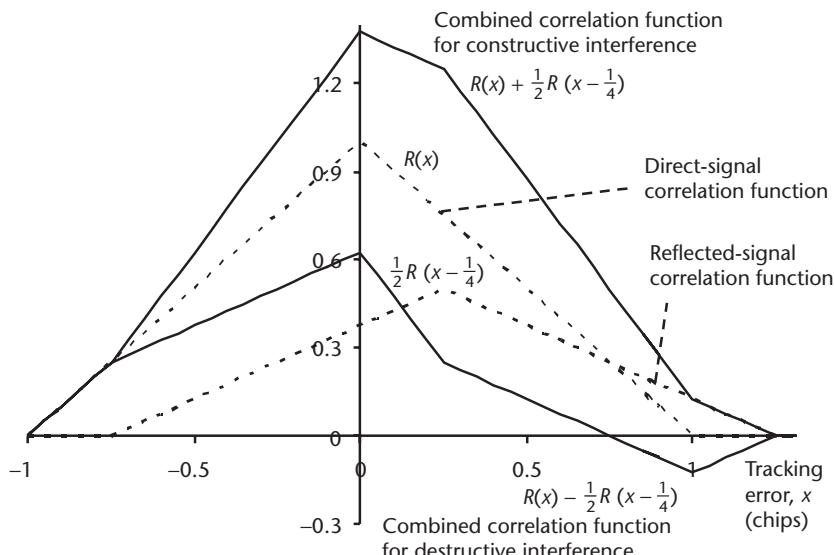


Figure 7.27 Direct, reflected, and combined signal correlation functions for $\delta = 1/4$, $\alpha = 1/2$, and $\delta\phi_{mi} = 0$ and 180° (neglecting precorrelation band-limiting).

As Figure 7.28 shows, the tracking error depends on the early-late correlator spacing. Multipath has less impact on the peak of the correlation function, so a narrower correlator spacing often leads to a smaller tracking error [48]. However, where precorrelation band-limiting is significant, the correlation function is rounded, reducing the benefit of narrowing the correlator spacing, as Figure 7.29 illustrates.

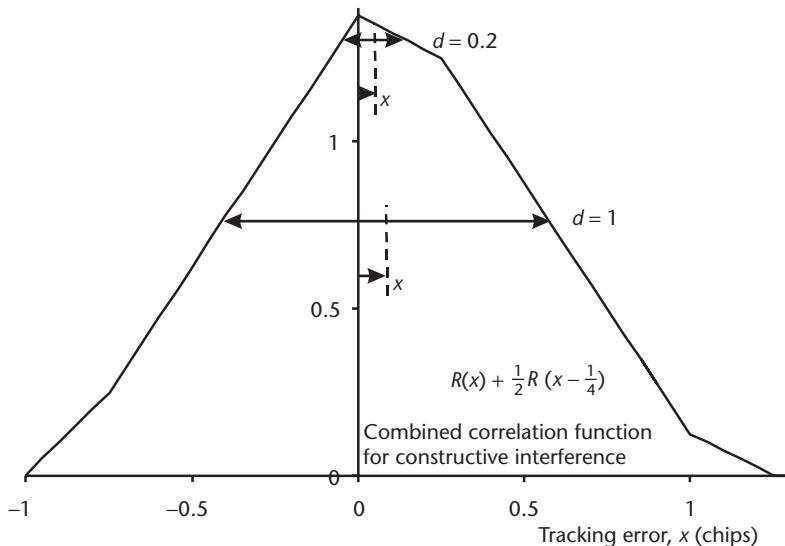


Figure 7.28 Effect of early-late correlator spacing on multipath error (neglecting precorrelation band-limiting).

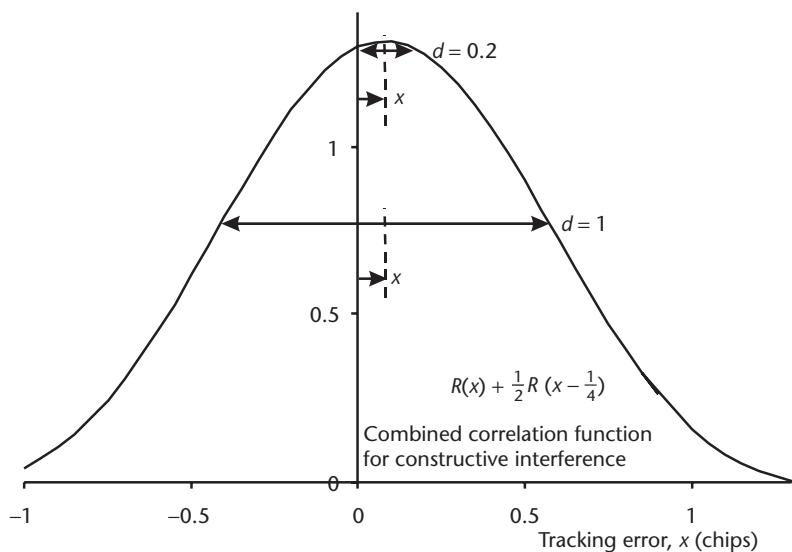


Figure 7.29 Effect of early-late correlator spacing on multipath error with a precorrelation band-width of $B_{PC} = 2f_{co}$.

An analytical solution to (7.166) is possible where there is a single delayed signal, the lag is small, the frequency offset is negligible, and precorrelation band-limiting may be neglected:

$$\begin{aligned} x &= \frac{\alpha^2 + \alpha \cos \delta\phi_m}{\alpha^2 + 2\alpha \cos \delta\phi_m + 1} \delta & (7.167) \\ &\quad |x - \delta| < d/2 \\ \delta\rho_m &= \frac{\alpha^2 + \alpha \cos \delta\phi_m}{\alpha^2 + 2\alpha \cos \delta\phi_m + 1} \Delta \end{aligned}$$

Otherwise, numerical methods must be used.

Figure 7.30 shows the limits of the code tracking error for different correlator spacings, assuming a BPSK signal [51]. The actual tracking error oscillates as the carrier phase offset changes. Note that the mean tracking error, averaged over the carrier phase offset, is nonzero. For BOC signals, the code tracking error exhibits $2f_s/f_{co} - 1$ nodes, evenly distributed in range error [31, 52].

Where the range lag is several chips, tracking errors of up to a meter can be caused by interference from one of the minor peaks of the GPS or GLONASS C/A code correlation function [53].

Multipath interference also produces carrier phase tracking errors. Where the user moves with respect to the reflecting surface, significant pseudo-range-rate/Doppler tracking errors are also seen.

Multipath mitigation techniques are discussed in Section 8.4.

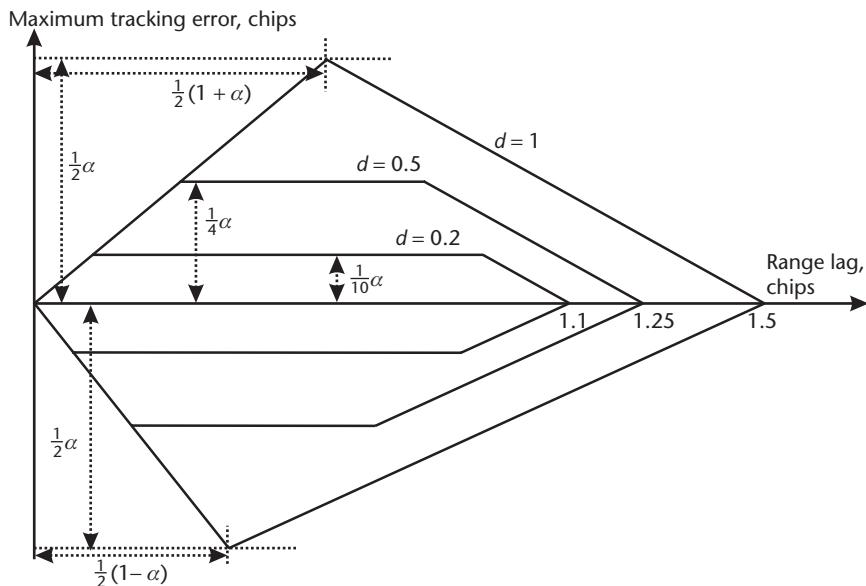


Figure 7.30 Limits of code tracking error due to multipath (neglecting precorrelation band-limiting).

7.5 Navigation Processor

The navigation processor calculates the user position and velocity solution and calibrates the receiver clock errors. The clock offset may be fed back to the ranging processor to correct the receiver clock, either on every iteration or when it exceeds a certain threshold, such as 1 ms. The clock drift may also be fed back.

The ranging processor outputs pseudo-range and pseudo-range-rate (or equivalent) measurements for all the satellites tracked at a common time of signal arrival. Note that the processing of carrier phase measurements is discussed in Section 8.2. Where the ranging processor does not apply corrections for the satellite clock errors and the ionosphere and troposphere propagation delays (Section 7.4), these corrections should be applied by the navigation processor.

The navigation processor estimates the user position, $\hat{\mathbf{r}}_{ia}^i = (\hat{x}_{ia}^i, \hat{y}_{ia}^i, \hat{z}_{ia}^i)$, and receiver clock offset, $\delta\hat{\rho}_{rc}$, at the time of signal arrival, t_{sa} . Each corrected pseudo-range measurement, $\tilde{\rho}_{Cj}$, may be expressed in terms of these estimates by

$$\begin{aligned}\tilde{\rho}_{Cj} &= \sqrt{[\hat{\mathbf{r}}_{isj}^i(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ia}^i(t_{sa})]^T [\hat{\mathbf{r}}_{isj}^i(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ia}^i(t_{sa})] + \delta\hat{\rho}_{rc}(t_{sa}) + \delta\rho_{\epsilon j}^+} \\ &= \sqrt{[\hat{x}_{isj}^i(\tilde{t}_{st,j}) - \hat{x}_{ia}^i(t_{sa})]^2 + [\hat{y}_{isj}^i(\tilde{t}_{st,j}) - \hat{y}_{ia}^i(t_{sa})]^2 + [\hat{z}_{isj}^i(\tilde{t}_{st,j}) - \hat{z}_{ia}^i(t_{sa})]^2} \\ &\quad + \delta\hat{\rho}_{rc}(t_{sa}) + \delta\rho_{\epsilon j}^+\end{aligned}\tag{7.168}$$

where $\hat{\mathbf{r}}_{isj}^i$ is the satellite position obtained from the navigation data message as described in Section 7.1.1, $\tilde{t}_{st,j}$ is the measured transmission time, and $\delta\rho_{\epsilon j}^+$ is the measurement residual.

The transmission time may be output by the ranging processor. Otherwise, it must be estimated from the arrival time and the range estimate, $\hat{\rho}_T$, given by (7.45), using (6.1). This requires either an iterative solution or use of the range estimate from the previous navigation processing cycle.

The measurement residual is, with reference to (7.44),

$$\delta\rho_{\epsilon j}^+ = \tilde{\rho}_{Cj} - \hat{\rho}_{Cj}\tag{7.169}$$

noting that it is only nonzero in an overdetermined or filtered navigation solution.

Similarly, each pseudo-range-rate measurement, $\tilde{\rho}_{Cj}$, is expressed in terms of the navigation processor's user velocity, $\hat{\mathbf{v}}_{ia}^i$, and receiver clock drift, $\delta\hat{\rho}_{rc}$, estimates by

$$\tilde{\rho}_{Cj} = \mathbf{u}_{as,j}^i T [\hat{\mathbf{v}}_{isj}^i(\tilde{t}_{st,j}) - \hat{\mathbf{v}}_{ia}^i(t_{sa})] + \delta\hat{\rho}_{rc}(t_{sa}) + \delta\rho_{\epsilon j}^+\tag{7.170}$$

where calculation of the satellite velocity, $\hat{\mathbf{v}}_{isj}^i$, and the line-of-sight vector, $\mathbf{u}_{as,j}^i$, are described in Section 7.1.1 and the measurement residual is

$$\delta\rho_{\epsilon j}^+ = \tilde{\rho}_{Cj} - \hat{\rho}_{Cj}\tag{7.171}$$

Where the user position and velocity are estimated in the ECEF frame, (7.168) and (7.170) are replaced by

$$\begin{aligned}\tilde{\rho}_{Cj} - \delta\rho_{ie,j} &= \sqrt{\left[\hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ea}^e(t_{sa}) \right]^T \left[\hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ea}^e(t_{sa}) \right]} + \delta\hat{\rho}_{rc}(t_{sa}) + \delta\rho_{\epsilon j}^+ \\ \tilde{\rho}_{Cj} - \delta\dot{\rho}_{ie,j} &= \mathbf{u}_{as,j}^e \left[\hat{\mathbf{v}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{v}}_{ea}^e(t_{sa}) \right] + \delta\hat{\rho}_{rc}(t_{sa}) + \delta\rho_{\epsilon j}^+\end{aligned}\quad (7.172)$$

where the Sagnac corrections are given by (7.27) and (7.40), or by

$$\begin{aligned}\tilde{\rho}_{Cj} &= \sqrt{\left[\mathbf{C}_e^I(\tilde{t}_{st,j}) \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ea}^e(t_{sa}) \right]^T \left[\mathbf{C}_e^I(\tilde{t}_{st,j}) \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ea}^e(t_{sa}) \right]} + \delta\hat{\rho}_{rc}(t_{sa}) + \delta\rho_{\epsilon j}^+ \\ \tilde{\rho}_{Cj} &= \mathbf{u}_{as,j}^e \left[\mathbf{C}_e^I(\tilde{t}_{st,j}) \left(\hat{\mathbf{v}}_{esj}^e(\tilde{t}_{st,j}) + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) \right) - \left(\hat{\mathbf{v}}_{ea}^e(t_{sa}) + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{r}}_{ea}^e(t_{sa}) \right) \right] \\ &\quad + \delta\hat{\rho}_{rc}(t_{sa}) + \delta\rho_{\epsilon j}^+\end{aligned}\quad (7.173)$$

The navigation solution is obtained by solving these equations with pseudo-range and pseudo-range-rate measurements from at least four satellites. A *single-point* or *snapshot* navigation solution uses only the current set of ranging processor measurements and is described first. A filtered navigation solution, described next, also makes use of previous measurement data. The filtered navigation solution is much less noisy but can exhibit dynamic-response lags, while successive single-point solutions are independent so can highlight erroneous measurement data more quickly. Thus, the single-point solution is useful for integrity monitoring (see Chapter 15). The accuracy of the single-point solution can be improved by using carrier-smoothed pseudo-range measurements, but the noise on successive solutions is then not independent. A single-point solution is also needed to initialize the filtered solution.

The section concludes with discussions of combined navigation and tracking algorithms and navigation error budgets.

7.5.1 Single-Point Navigation Solution

A position solution cannot easily be obtained analytically from a set of pseudo-range measurements using (7.168). Therefore, the equations are linearized by performing a Taylor expansion about a predicted user position, \mathbf{r}_{ia}^{ip} , and clock offset, $\delta\rho_{rc}^p$, in analogy with the linearized Kalman filter (Section 3.4.1). The predicted user position and clock offset is generally the solution from the previous set of pseudo-range measurements. At initialization, the solution may have to be iterated a number of times. Thus, (7.168) is replaced by

$$\begin{pmatrix} \tilde{\rho}_{C1} - \rho_{C1}^p \\ \tilde{\rho}_{C2} - \rho_{C2}^p \\ \vdots \\ \tilde{\rho}_{Cn} - \rho_{Cn}^p \end{pmatrix} = \mathbf{H} \begin{pmatrix} \hat{x}_{ia}^i(t_{sa}) - x_{ia}^{ip} \\ \hat{y}_{ia}^i(t_{sa}) - y_{ia}^{ip} \\ \hat{z}_{ia}^i(t_{sa}) - z_{ia}^{ip} \\ \delta\hat{\rho}_{rc}(t_{sa}) - \delta\rho_{rc}^p \end{pmatrix} + \begin{pmatrix} \delta\rho_{\epsilon 1}^+ \\ \delta\rho_{\epsilon 2}^+ \\ \vdots \\ \delta\rho_{\epsilon n}^+ \end{pmatrix} \quad (7.174)$$

where the number of measurements, n , is at least four . The predicted pseudo-ranges are given by

$$\rho_{Cj}^P = \sqrt{[\hat{\mathbf{r}}_{isj}^i(\tilde{t}_{st,j}) - \mathbf{r}_{ia}^{iP}]^T [\hat{\mathbf{r}}_{isj}^i(\tilde{t}_{st,j}) - \mathbf{r}_{ia}^{iP}]} + \delta\rho_{rc}^P \quad (7.175)$$

and the measurement matrix, \mathbf{H} , is

$$\mathbf{H} = \begin{pmatrix} \frac{\partial\rho_1}{\partial x_{ia}^i} & \frac{\partial\rho_1}{\partial y_{ia}^i} & \frac{\partial\rho_1}{\partial z_{ia}^i} & \frac{\partial\rho_1}{\partial \rho_{rc}} \\ \frac{\partial\rho_2}{\partial x_{ia}^i} & \frac{\partial\rho_2}{\partial y_{ia}^i} & \frac{\partial\rho_2}{\partial z_{ia}^i} & \frac{\partial\rho_2}{\partial \rho_{rc}} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial\rho_n}{\partial x_{ia}^i} & \frac{\partial\rho_n}{\partial y_{ia}^i} & \frac{\partial\rho_n}{\partial z_{ia}^i} & \frac{\partial\rho_n}{\partial \rho_{rc}} \end{pmatrix} \mid_{\mathbf{r}_{ia}^i = \mathbf{r}_{ia}^{iP}} \quad (7.176)$$

Differentiating (7.168) with respect to the user position and clock offset gives

$$\mathbf{H} = \begin{pmatrix} -u_{as,1,x}^{iP} & -u_{as,1,y}^{iP} & -u_{as,1,z}^{iP} & 1 \\ -u_{as,2,x}^{iP} & -u_{as,2,y}^{iP} & -u_{as,2,z}^{iP} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -u_{as,n,x}^{iP} & -u_{as,n,y}^{iP} & -u_{as,n,z}^{iP} & 1 \end{pmatrix} = \mathbf{G}_i \quad (7.177)$$

where the line-of-sight unit vectors are obtained from (7.34) using the predicted user position. From (7.50), the measurement matrix is the same as the geometry matrix, \mathbf{G} . Where there are four pseudo-range measurements, the number of measurements matches the number of unknowns, so the measurement residuals are zero. The position and clock solution is then

$$\begin{pmatrix} \hat{\mathbf{r}}_{ia}^i(t_{sa}) \\ \hat{\delta\rho}_{rc}(t_{sa}) \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{ia}^{iP} \\ \delta\rho_{rc}^P \end{pmatrix} + \mathbf{G}_i^{-1} \begin{pmatrix} \tilde{\rho}_{C1} - \rho_{C1}^P \\ \tilde{\rho}_{C2} - \rho_{C2}^P \\ \tilde{\rho}_{C3} - \rho_{C3}^P \\ \tilde{\rho}_{C4} - \rho_{C4}^P \end{pmatrix} \quad (7.178)$$

Where there are more than four pseudo-range measurements, the solution is overdetermined and, without the measurement residual terms, the set of measurements would not produce a consistent navigation solution. However, the extra measurements provide the opportunity to smooth out some of the measurement noise. This is done by minimizing the sum of squares of the residuals. Thus,

$$\frac{\partial}{\partial (\hat{\mathbf{r}}_{ia}^i(t_{sa}), \hat{\delta\rho}_{rc}(t_{sa}))} \sum_{j=1}^n \delta\rho_{\epsilon_j}^{+2} = 0 \quad (7.179)$$

Applying this to (7.174) gives the least-squares solution [54]:

$$\begin{pmatrix} \hat{\mathbf{r}}_{ia}^i(t_{sa}) \\ \delta\hat{\rho}_{rc}(t_{sa}) \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{ia}^{iP} \\ \delta\rho_{rc}^P \end{pmatrix} + (\mathbf{G}_i^T \mathbf{G}_i)^{-1} \mathbf{G}_i^T \begin{pmatrix} \tilde{\rho}_{C1} - \rho_{C1}^P \\ \tilde{\rho}_{C2} - \rho_{C2}^P \\ \vdots \\ \tilde{\rho}_{Cn} - \rho_{Cn}^P \end{pmatrix} \quad (7.180)$$

Where the accuracy of the pseudo-range measurements is known to differ, for example due to variation in c/n_0 or the residual ionosphere and troposphere propagation errors, which depend on the elevation angle, a weighted least-squares estimate can be computed [54]:

$$\begin{pmatrix} \hat{\mathbf{r}}_{ia}^i(t_{sa}) \\ \delta\hat{\rho}_{rc}(t_{sa}) \end{pmatrix} = \begin{pmatrix} \mathbf{r}_{ia}^{iP} \\ \delta\rho_{rc}^P \end{pmatrix} + (\mathbf{G}_i^T \mathbf{C}^{+1} \mathbf{G}_i)^{-1} \mathbf{G}_i^T \mathbf{C}^{+1} \begin{pmatrix} \tilde{\rho}_{C1} - \rho_{C1}^P \\ \tilde{\rho}_{C2} - \rho_{C2}^P \\ \vdots \\ \tilde{\rho}_{Cn} - \rho_{Cn}^P \end{pmatrix} \quad (7.181)$$

where the diagonal elements of the measurement residual covariance matrix, \mathbf{C}^+ , are the predicted variances of each pseudo-range error and the off-diagonal terms account for any correlations between the pseudo-range errors. Note that the residual covariance matrix includes time-correlated errors (i.e., biases), while the Kalman filter measurement noise covariance matrix, \mathbf{R} , does not.

The least-squares velocity and receiver clock drift solution is

$$\begin{pmatrix} \hat{\mathbf{v}}_{ia}^i(t_{sa}) \\ \delta\hat{\rho}_{rc}(t_{sa}) \end{pmatrix} = \begin{pmatrix} \mathbf{v}_{ia}^{iP} \\ \delta\rho_{rc}^P \end{pmatrix} + (\mathbf{G}_i^T \mathbf{G}_i)^{-1} \mathbf{G}_i^T \begin{pmatrix} \tilde{\rho}_{C1} - \dot{\rho}_{C1}^P \\ \tilde{\rho}_{C2} - \dot{\rho}_{C2}^P \\ \vdots \\ \tilde{\rho}_{Cn} - \dot{\rho}_{Cn}^P \end{pmatrix} \quad (7.182)$$

where

$$\dot{\rho}_{Cj}^P = \mathbf{u}_{as,j}^{iP} {}^T [\hat{\mathbf{v}}_{isj}^i(\tilde{t}_{st,j}) - \mathbf{v}_{ia}^{iP}] + \delta\dot{\rho}_{rc}^P \quad (7.183)$$

and noting that the measurement matrix is the same.

Where the ECEF frame is used, the least-squares solution is

$$\begin{aligned} \begin{pmatrix} \hat{\mathbf{r}}_{ea}^e(t_{sa}) \\ \hat{\delta\rho}_{rc}^e(t_{sa}) \end{pmatrix} &= \begin{pmatrix} \mathbf{r}_{ea}^{eP} \\ \delta\rho_{rc}^P \end{pmatrix} + (\mathbf{G}_e^T \mathbf{G}_e)^{-1} \mathbf{G}_e^T \begin{pmatrix} \tilde{\rho}_{C1} - \rho_{C1}^P \\ \tilde{\rho}_{C2} - \rho_{C2}^P \\ \vdots \\ \tilde{\rho}_{Cn} - \rho_{Cn}^P \end{pmatrix} \quad (7.184) \\ \begin{pmatrix} \hat{\mathbf{v}}_{ea}^e(t_{sa}) \\ \hat{\delta\dot{\rho}}_{rc}^e(t_{sa}) \end{pmatrix} &= \begin{pmatrix} \mathbf{v}_{ea}^{eP} \\ \delta\dot{\rho}_{rc}^P \end{pmatrix} + (\mathbf{G}_e^T \mathbf{G}_e)^{-1} \mathbf{G}_e^T \begin{pmatrix} \tilde{\dot{\rho}}_{C1} - \dot{\rho}_{C1}^P \\ \tilde{\dot{\rho}}_{C2} - \dot{\rho}_{C2}^P \\ \vdots \\ \tilde{\dot{\rho}}_{Cn} - \dot{\rho}_{Cn}^P \end{pmatrix} \end{aligned}$$

where

$$\rho_{Cj}^P = \sqrt{[\hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \mathbf{r}_{ea}^{eP}]^T [\hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \mathbf{r}_{ea}^{eP}] + \delta\rho_{rc}^P + \delta\rho_{ie,j}} \quad (7.185)$$

$$\dot{\rho}_{Cj}^P = \mathbf{u}_{as,j}^{eP T} [\hat{\mathbf{v}}_{esj}^e(\tilde{t}_{st,j}) - \mathbf{v}_{ea}^{eP}] + \delta\dot{\rho}_{rc}^P + \delta\rho_{ie,j}$$

or

$$\begin{aligned} \rho_{Cj}^P &= \sqrt{[\mathbf{C}_e^I(\tilde{t}_{st,j}) \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \mathbf{r}_{ea}^{eP}]^T [\mathbf{C}_e^I(\tilde{t}_{st,j}) \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \mathbf{r}_{ea}^{eP}] + \delta\rho_{rc}^P} \\ \dot{\rho}_{Cj}^P &= \mathbf{u}_{as,j}^{e T} [\mathbf{C}_e^I(\tilde{t}_{st,j}) (\hat{\mathbf{v}}_{esj}^e(\tilde{t}_{st,j}) + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j})) - (\mathbf{v}_{ea}^{eP} + \boldsymbol{\Omega}_{ie}^e \mathbf{r}_{ea}^{eP})] + \delta\dot{\rho}_{rc}^P \end{aligned} \quad (7.186)$$

It is easier to obtain the curvilinear position, (L_a, λ_a, h_a) and the velocity in local navigation frame axes, \mathbf{v}_{ea}^n , from \mathbf{r}_{ea}^e , and \mathbf{v}_{ea}^e using (2.71) and (2.41), where \mathbf{C}_e^n is given by (2.99), than to calculate them directly.

7.5.2 Filtered Navigation Solution

The main disadvantage of the single-point navigation solution is that it discards useful information from previous measurements. Specifically, the prior clock offset and drift measurements provide a good indication of the current clock offset, and the prior position and velocity measurements provide a good indication of the current position.¹

Therefore, most GNSS user equipment uses a *navigation filter*. This maintains continuous estimates of the position, velocity, clock bias, and clock drift and uses the pseudo-range and pseudo-range-rate measurements to correct them. The velocity estimates are used to update the position estimates, and the clock drift is used to update the clock offset.²

A Kalman filter-based estimation algorithm, described in Chapter 3, is used to maintain optimum weighting of the current set of pseudo-range and pseudo-

range-rate measurements against the estimates obtained from previous measurements.

The filtered navigation solution has a number of advantages. The carrier-derived pseudo-range-rate measurements smooth out the code tracking noise on the position solution. A navigation solution can be maintained for a limited period with only three satellites where the clock errors are well calibrated and a rough navigation solution can be maintained for a few seconds when all GNSS signals are blocked, such as in tunnels.

The choice of states to estimate is now discussed, followed by descriptions of the system and measurement models and a discussion of the handling of range biases, constellation changes, and ephemeris updates. The Kalman filter algorithm is described in Sections 3.2.2 and 3.4.1.

7.5.2.1 State Selection

The GNSS navigation solution comprises the Kalman filter state vector. Finding the best states to estimate depends on the application [55, 56].

For static applications, such as surveying, the position, \mathbf{r} , is estimated, but the velocity need not be. For low-dynamics applications, such as handheld receivers, land vehicles, and marine applications, the velocity, \mathbf{v} , must be estimated in addition to the position, but the effects of acceleration may be modeled as system noise on the velocity estimates. For high-dynamics applications, such as fighter aircraft, missiles, and space launch vehicles, the acceleration must also be estimated in a GNSS navigation filter. However, an integrated INS/GNSS navigation system (Chapter 12) is generally used in practice.¹

The navigation states may be implemented in any coordinate frame. An ECI-frame implementation leads to the simplest system and measurement models. Estimating latitude, longitude, and height with Earth-referenced velocity in local navigation frame axes avoids the need to apply a coordinate frame transformation to the user outputs, while a Cartesian ECEF-frame implementation is very common.

The receiver clock offset, $\delta\rho_{rc}$, and drift, $\delta\dot{\rho}_{rc}$, must always be estimated. For very-high-dynamics applications, modeling the clock g-dependent error, either as states or correlated system noise (Section 3.4.2), should be considered.

The other significant parameters in the GNSS navigation problem are the correlated range errors due to a combination of the ephemeris and the residual satellite clock, ionosphere, and troposphere errors (see Section 7.4). These have standard deviations of a few meters and correlation times of around 30 minutes. They are difficult to observe as Kalman filter states. If only four satellites are used to form the navigation solution, the range biases are unobservable. Where all satellites in view are used, the range biases are partially observable. Their inclusion in the state vector improves the accuracy of the other estimates and produces a more representative error covariance matrix, \mathbf{P} . Range-bias observation is improved in very low dynamics or with tightly coupled INS/GNSS integration. It is also improved by using a high-performance reference oscillator.²

Where a dual-frequency receiver is used, pseudo-range measurements on each frequency may be input separately and the ionosphere propagation delays estimated

as Kalman filter states, instead of using combined pseudo-ranges, in which case smoothing of the ionosphere corrections is implicit.

For the navigation filter system and measurement models described here, eight states are estimated: the antenna position and velocity and the receiver clock offset and drift. The ECI-frame implementation is described first, followed by discussions of the ECEF and local-navigation-frame variants. The state vectors are

$$\mathbf{x}^i = \begin{pmatrix} \mathbf{r}_{ia}^i \\ \mathbf{v}_{ia}^i \\ \delta\rho_{rc} \\ \dot{\delta\rho}_{rc} \end{pmatrix}, \quad \mathbf{x}^e = \begin{pmatrix} \mathbf{r}_{ea}^e \\ \mathbf{v}_{ea}^e \\ \delta\rho_{rc} \\ \dot{\delta\rho}_{rc} \end{pmatrix}, \quad \mathbf{x}^n = \begin{pmatrix} L_a \\ \lambda_a \\ h_a \\ \mathbf{v}_{ea}^e \\ \delta\rho_{rc} \\ \dot{\delta\rho}_{rc} \end{pmatrix} \quad (7.187)$$

where the superscripts i , e , and n are used to distinguish between the implementations in the different coordinate frames.

7.5.2.2 System Model

The Kalman filter system model (Section 3.2.4) describes how the states and their uncertainties are propagated forward in time to account for the user motion and receiver clock dynamics between successive measurements from the GNSS ranging processor. It also maintains a rough navigation solution during signal outages.

The system model for GNSS navigation in the ECI frame is simple. From (2.35), the time derivative of the position is the velocity, while the time derivative of the clock offset is the clock drift. The velocity and clock drift are not functions of any of the Kalman filter states. The state dynamics are thus

$$\dot{\mathbf{r}}_{ia}^i = \mathbf{v}_{ia}^i, \quad \frac{\partial}{\partial t} \delta\rho_{rc} = \dot{\delta\rho}_{rc} \quad (7.188)$$

Substituting this into (3.18) gives the system matrix:

$$\mathbf{F}^i = \begin{pmatrix} 0 & 0 & 0 & | & 1 & 0 & 0 & | & 0 & | & 0 \\ 0 & 0 & 0 & | & 0 & 1 & 0 & | & 0 & | & 0 \\ 0 & -0 & -0 & | & 0 & -0 & -0 & | & -1 & | & 0 & | & 0 \\ 0 & 0 & 0 & | & 0 & 0 & 0 & | & 0 & | & 0 & | & 0 \\ 0 & 0 & 0 & | & 0 & 0 & 0 & | & 0 & | & 0 & | & 0 \\ 0 & -0 & -0 & | & 0 & -0 & -0 & | & 0 & | & 0 & | & 0 \\ 0 & 0 & 0 & | & 0 & 0 & 0 & | & 0 & | & 0 & | & 1 \\ 0 & 0 & 0 & | & 0 & 0 & 0 & | & 0 & | & 0 & | & 0 \end{pmatrix} \quad (7.189)$$

It is convenient to express the Kalman filter matrices in terms of submatrices corresponding to the vector subcomponents of the state vector \mathbf{r}_{ia}^i , \mathbf{v}_{ia}^i , $\Delta\rho_{rc}$, and $\dot{\Delta\rho}_{rc}$. Thus,

$$\mathbf{F}^i = \begin{pmatrix} 0_3 & \mathbf{I}_3 & 0_{3,1} & 0_{3,1} \\ 0_3 & 0_3 & 0_{3,1} & 0_{3,1} \\ 0_{1,3} & 0_{1,3} & 0 & 1 \\ 0_{1,3} & 0_{1,3} & 0 & 0 \end{pmatrix} \quad (7.190)$$

where \mathbf{I}_n is the $n \times n$ identity matrix, 0_n is the $n \times n$ null matrix, and $0_{n,m}$ is the $n \times m$ null matrix.³

The state dynamics for the ECEF-frame implementation are the same, so $\mathbf{F}^e = \mathbf{F}^i$. Where the local-navigation-frame implementation with curvilinear position is used, the time derivative of the position is given by (2.72). Strictly, this violates the linearity assumption of the Kalman filter system model. However, the denominators may be treated as constant over the state propagation interval, so

$$\mathbf{F}^n \approx \begin{pmatrix} 0_3 & \mathbf{F}_{12}^n & 0_{3,1} & 0_{3,1} \\ 0_3 & 0_3 & 0_{3,1} & 0_{3,1} \\ 0_{1,3} & 0_{1,3} & 0 & 1 \\ 0_{1,3} & 0_{1,3} & 0 & 0 \end{pmatrix} \quad (7.191)$$

$$\mathbf{F}_{12}^n = \begin{pmatrix} 1/[R_N(\hat{L}_a) + \hat{h}_a] & 0 & 0 \\ 0 & 1/[(R_E(\hat{L}_a) + \hat{h}_a)\cos \hat{L}_a] & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

where R_N and R_E are given by (2.65) and (2.66).

The higher order terms in (3.25) are zero for all three implementations, so the transition matrix is simply

$$\Phi_k = \mathbf{I}_8 + \mathbf{F}_k \tau_s \quad (7.192)$$

where τ_s is the state propagation interval.

The main sources of increased uncertainty over the state integration interval are changes in velocity due to user motion and the random walk of the receiver clock drift. The system noise covariance matrix is thus

$$\mathbf{Q} = \begin{pmatrix} 0_3 & 0_3 & 0_{3,1} & 0_{3,1} \\ 0_3 & \mathbf{Q}_{22} & 0_{3,1} & 0_{3,1} \\ 0_{1,3} & 0_{1,3} & 0 & 0 \\ 0_{1,3} & 0_{1,3} & 0 & n_{rca}^2 \tau_s \end{pmatrix} \quad (7.193)$$

$$\mathbf{Q}_{22}^i = \mathbf{Q}_{22}^e = \begin{pmatrix} n_{a,x}^2 & 0 & 0 \\ 0 & n_{a,y}^2 & 0 \\ 0 & 0 & n_{a,z}^2 \end{pmatrix} \tau_s \quad \mathbf{Q}_{22}^n = \begin{pmatrix} n_{a,N}^2 & 0 & 0 \\ 0 & n_{a,E}^2 & 0 \\ 0 & 0 & n_{a,D}^2 \end{pmatrix} \tau_s$$

where n_a^2 is the acceleration PSD and n_{rca}^2 is the receiver clock drift-rate PSD. These are

$$n_{a,i}^2 = \frac{\sigma^2 (\nu_{\beta a,i}^\gamma(t + \tau_s) - \nu_{\beta a,i}^\gamma(t))}{\tau_s}, \quad i \in x, y, z, N, E, D, \quad \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (7.194)$$

and

$$n_{rca}^2 = \frac{\sigma^2 (\delta\dot{p}_{rc}(t + \tau_s) - \delta\dot{p}_{rc}(t))}{\tau_s} \quad (7.195)$$

The acceleration PSD depends on the dynamics of the application and is usually greater in the horizontal axes than the vertical. For a car, $n_{a,N}^2 = n_{a,E}^2 = 10 \text{ m}^2 \text{ s}^{-3}$ is reasonable. Where velocity is not estimated, system noise must be modeled on the position states to keep the Kalman filter receptive to new measurements. The clock drift-rate PSD can vary up to $0.04 \text{ m}^2 \text{ s}^{-3}$ [56], depending on the quality of the reference oscillator.

In a GNSS navigation filter, the measurement update rate can be as low as 1 Hz. The system noise covariance, \mathbf{Q} , can make a substantial contribution to the error covariance, \mathbf{P} , over this interval. So, either the first order in $\Phi \mathbf{Q} \Phi^T$ form of the error covariance propagation, (3.32), should be used or the system-propagation phase of the Kalman filter should be iterated at a faster rate than the measurement-update phase.¹

GNSS navigation is a total-state implementation of the Kalman filter, so a nonzero initialization of the state estimates is required. A single-point navigation solution (Section 7.5.1) can be used to initialize the position and clock offset, while, for most applications, it is sufficient to initialize the velocity to that of the Earth at the initial position and the clock drift estimate to zero.

The initial values of the error covariance matrix, \mathbf{P} , must reflect the precision of the position and clock-offset initialization process and the standard deviations of the host vehicle's velocity, with respect to the Earth, and the receiver clock drift.²

7.5.2.3 Measurement Model

The measurement model (Section 3.2.5) of a GNSS navigation filter updates the navigation solution using the measurements from the ranging processor and is analogous to the single-point solution described in Section 7.5.1. The measurement vector comprises the pseudo-ranges and pseudo-range rates output by the navigation processor. Thus, for n satellites tracked,

$$\mathbf{z}_G = (\tilde{p}_{C1}, \tilde{p}_{C2}, \dots, \tilde{p}_{Cn}, \mid \tilde{\dot{p}}_{C1}, \tilde{\dot{p}}_{C2}, \dots, \tilde{\dot{p}}_{Cn}) \quad (7.196)$$

where the subscript G denotes a GNSS measurement.

The pseudo-ranges and pseudo-range rates, modeled by (7.168) to (7.172), are not linear functions of the state estimates. Therefore, an extended Kalman filter

measurement model (Section 3.4.1) must be used. The measurement innovation vector is given by

$$\delta \mathbf{z}_{G,k}^- = \mathbf{z}_{G,k} - \mathbf{h}_G(\hat{\mathbf{x}}_k^-) \quad (7.197)$$

where

$$\mathbf{h}_G(\hat{\mathbf{x}}_k^-) = (\hat{\rho}_{C1}^-, \hat{\rho}_{C2}^-, \dots, \hat{\rho}_{Cn}^-, | \hat{\dot{\rho}}_{C1}^-, \hat{\dot{\rho}}_{C2}^-, \dots, \hat{\dot{\rho}}_{Cn}^-)_k \quad (7.198)$$

The predicted pseudo-ranges and pseudo-range rates are the same as in the single-point solution except that the predicted user position and velocity and receiver clock offset and drift are replaced by the Kalman filter estimates, propagated forward using the system model. Thus, in the ECI-frame implementation,

$$\hat{\rho}_{Cj,k}^- = \sqrt{[\hat{\mathbf{r}}_{isj}^i(\tilde{t}_{st,j,k}) - \hat{\mathbf{r}}_{ia,k}^{i-}]^T [\hat{\mathbf{r}}_{isj}^i(\tilde{t}_{st,j,k}) - \hat{\mathbf{r}}_{ia,k}^{i-}]} + \delta\hat{\rho}_{rc,k}^- \quad (7.199)$$

$$\hat{\dot{\rho}}_{Cj,k}^- = \hat{\mathbf{u}}_{as,j,k}^T [\hat{\mathbf{v}}_{isj}^i(\tilde{t}_{st,j,k}) - \hat{\mathbf{v}}_{ia,k}^{i-}] + \delta\hat{\dot{\rho}}_{rc,k}^-$$

where the line-of-sight unit vector is obtained from (7.34) using $\hat{\mathbf{r}}_{ia,k}^{i-}$. From (3.56), the measurement matrix is

$$\mathbf{H}_{G,k}^i = \left[\begin{array}{cccccc|c} \frac{\partial \rho_1}{\partial x_{ia}^i} & \frac{\partial \rho_1}{\partial y_{ia}^i} & \frac{\partial \rho_1}{\partial z_{ia}^i} & 0 & 0 & 0 & \frac{\partial \rho_1}{\partial \rho_{rc}} & 0 \\ \frac{\partial \rho_2}{\partial x_{ia}^i} & \frac{\partial \rho_2}{\partial y_{ia}^i} & \frac{\partial \rho_2}{\partial z_{ia}^i} & 0 & 0 & 0 & \frac{\partial \rho_2}{\partial \rho_{rc}} & 0 \\ \vdots & \vdots \\ \frac{\partial \rho_n}{\partial x_{ia}^i} & \frac{\partial \rho_n}{\partial y_{ia}^i} & \frac{\partial \rho_n}{\partial z_{ia}^i} & 0 & 0 & 0 & \frac{\partial \rho_n}{\partial \rho_{rc}} & 0 \\ \hline \frac{\partial \dot{\rho}_1}{\partial x_{ia}^i} & \frac{\partial \dot{\rho}_1}{\partial y_{ia}^i} & \frac{\partial \dot{\rho}_1}{\partial z_{ia}^i} & \frac{\partial \dot{\rho}_1}{\partial v_{ia,x}^i} & \frac{\partial \dot{\rho}_1}{\partial v_{ia,y}^i} & \frac{\partial \dot{\rho}_1}{\partial v_{ia,z}^i} & 0 & \frac{\partial \dot{\rho}_1}{\partial \dot{\rho}_{rc}} \\ \frac{\partial \dot{\rho}_2}{\partial x_{ia}^i} & \frac{\partial \dot{\rho}_2}{\partial y_{ia}^i} & \frac{\partial \dot{\rho}_2}{\partial z_{ia}^i} & \frac{\partial \dot{\rho}_2}{\partial v_{ia,x}^i} & \frac{\partial \dot{\rho}_2}{\partial v_{ia,y}^i} & \frac{\partial \dot{\rho}_2}{\partial v_{ia,z}^i} & 0 & \frac{\partial \dot{\rho}_2}{\partial \dot{\rho}_{rc}} \\ \vdots & \vdots \\ \frac{\partial \dot{\rho}_n}{\partial x_{ia}^i} & \frac{\partial \dot{\rho}_n}{\partial y_{ia}^i} & \frac{\partial \dot{\rho}_n}{\partial z_{ia}^i} & \frac{\partial \dot{\rho}_n}{\partial v_{ia,x}^i} & \frac{\partial \dot{\rho}_n}{\partial v_{ia,y}^i} & \frac{\partial \dot{\rho}_n}{\partial v_{ia,z}^i} & 0 & \frac{\partial \dot{\rho}_n}{\partial \dot{\rho}_{rc}} \end{array} \right] \mathbf{x} = \hat{\mathbf{x}}_k^- \quad (7.200)$$

noting that the pseudo-ranges are not functions of the user velocity or clock drift, while the pseudo-range rates are not functions of the clock offset. The dependence of the pseudo-range rates on position is weak with a 1m position error having a similar impact to a $\sim 5 \times 10^{-5} \text{ m s}^{-1}$ velocity error, so the $\partial \dot{\rho} / \partial \mathbf{r}$ terms are commonly neglected. Thus, from (7.168) and (7.170)

$$\mathbf{H}_{G,k}^i \approx \begin{pmatrix} -u_{as,1,x}^i & -u_{as,1,y}^i & -u_{as,1,z}^i & 0 & 0 & 0 & 1 & 0 \\ -u_{as,2,x}^i & -u_{as,2,y}^i & -u_{as,2,z}^i & 0 & 0 & 0 & 1 & 0 \\ \vdots & \vdots \\ -u_{as,n,x}^i & -u_{as,n,y}^i & -u_{as,n,z}^i & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & -u_{as,1,x}^i & -u_{as,1,y}^i & -u_{as,1,z}^i & 0 & 1 \\ 0 & 0 & 0 & -u_{as,2,x}^i & -u_{as,2,y}^i & -u_{as,2,z}^i & 0 & 1 \\ \vdots & \vdots \\ 0 & 0 & 0 & -u_{as,n,x}^i & -u_{as,n,y}^i & -u_{as,n,z}^i & 0 & 1 \end{pmatrix} \mathbf{x} = \hat{\mathbf{x}}_k^- \quad (7.201)$$

For the Cartesian ECEF-frame implementation, the predicted pseudo-range rates are

$$\begin{aligned} \hat{\rho}_{Cj,k}^- &= \sqrt{[\hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j,k}) - \hat{\mathbf{r}}_{ea,k}^e]^T [\hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j,k}) - \hat{\mathbf{r}}_{ea,k}^e]} + \hat{\delta\rho}_{rc,k}^- + \delta\rho_{ie,j} \\ \hat{\rho}_{Cj,k}^- &= \hat{\mathbf{u}}_{as,j,k}^{e-} \left[\hat{\mathbf{v}}_{esj}^e(\tilde{t}_{st,j,k}) - \hat{\mathbf{v}}_{ea,k}^e \right] + \hat{\delta\rho}_{rc,k}^- + \delta\rho_{ie,j} \end{aligned} \quad (7.202)$$

or

$$\begin{aligned} \hat{\rho}_{Cj,k}^- &= \sqrt{[\mathbf{C}_e^I(\tilde{t}_{st,j,k}) \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j,k}) - \hat{\mathbf{r}}_{ea,k}^e]^T [\mathbf{C}_e^I(\tilde{t}_{st,j,k}) \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j,k}) - \hat{\mathbf{r}}_{ea,k}^e]} + \hat{\delta\rho}_{rc,k}^- \\ \hat{\rho}_{Cj,k}^- &= \hat{\mathbf{u}}_{as,j,k}^e \left[\mathbf{C}_e^I(\tilde{t}_{st,j,k}) (\hat{\mathbf{v}}_{esj}^e(\tilde{t}_{st,j,k}) + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j,k})) - (\hat{\mathbf{v}}_{ea,k}^e + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{r}}_{ea,k}^e) \right] + \hat{\delta\rho}_{rc,k}^- \end{aligned} \quad (7.203)$$

while the measurement matrix, $\mathbf{H}_{G,k}^e$, is as $\mathbf{H}_{G,k}^i$ with $\mathbf{u}_{as,j}^e$ substituted for $\mathbf{u}_{as,j}^i$.

For the local-navigation-frame implementation, it is easiest to compute the predicted pseudo-ranges and pseudo-range rates as above using the Cartesian position, calculated using (2.70), and ECEF velocity, calculated using (2.41). The measurement matrix is

$$\mathbf{H}_{G,k}^n \approx \begin{pmatrix} b_L u_{as,1,N}^n & b_\lambda u_{as,1,E}^n & u_{as,1,D}^n & 0 & 0 & 0 & 1 & 0 \\ b_L u_{as,2,N}^n & b_\lambda u_{as,2,E}^n & u_{as,2,D}^n & 0 & 0 & 0 & 1 & 0 \\ \vdots & \vdots \\ b_L u_{as,n,N}^n & b_\lambda u_{as,n,E}^n & u_{as,n,D}^n & 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & -u_{as,1,N}^n & -u_{as,1,E}^n & -u_{as,1,D}^n & 0 & 1 \\ 0 & 0 & 0 & -u_{as,2,N}^n & -u_{as,2,E}^n & -u_{as,2,D}^n & 0 & 1 \\ \vdots & \vdots \\ 0 & 0 & 0 & -u_{as,n,N}^n & -u_{as,n,E}^n & -u_{as,n,D}^n & 0 & 1 \end{pmatrix} \mathbf{x} = \hat{\mathbf{x}}_k^- \quad (7.204)$$

where

$$b_L = -[R_N(\hat{L}_a) + \hat{b}_a], \quad b_\lambda = -[R_E(\hat{L}_a) + \hat{h}_a] \cos \hat{L}_a \quad (7.205)$$

The measurement noise covariance matrix, \mathbf{R}_G , models the noise-like errors on the pseudo-range and pseudo-range-rate measurements, such as tracking errors, multipath variations, and satellite clock noise, but does not account for bias-like errors. In many GNSS navigation filters, \mathbf{R}_G is modeled as diagonal and constant, but there can be benefits in varying it as a function of \tilde{c}/\tilde{n}_0 and/or the level of dynamics, where known.

The pseudo-range and pseudo-range-rate measurements are generally uncorrelated with each other. However, if carrier-smoothed pseudo-range, (7.130) or (7.142), is used, correlations between the noise on the corresponding pseudo-range and pseudo-range-rate measurements must be modeled.³

7.5.2.4 Range Biases, Constellation Changes, and Ephemeris Updates

Where the correlated range errors due to residual ionosphere, troposphere, satellite clock, and ephemeris errors are not estimated by the Kalman filter, they will bias the position and clock offset estimates away from their true values. To account for this, an extra term should be added to the state uncertainty modeled by the Kalman filter. The corrected position and clock offset are then

$$\begin{pmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \sigma_{prc} \end{pmatrix} = \begin{pmatrix} \sqrt{P_{1,1} + \Delta \sigma_x^2} \\ \sqrt{P_{2,2} + \Delta \sigma_y^2} \\ \sqrt{P_{3,3} + \Delta \sigma_z^2} \\ \sqrt{P_{7,7} + \Delta \sigma_{prc}^2} \end{pmatrix} \quad (7.206)$$

for ECI and Cartesian ECEF frame position, where

$$\begin{pmatrix} \Delta \sigma_x \\ \Delta \sigma_y \\ \Delta \sigma_z \\ \Delta \sigma_{prc} \end{pmatrix} = \begin{pmatrix} D_x \\ D_y \\ D_z \\ D_T \end{pmatrix} \sigma_{prc} \quad (7.207)$$

and σ_{prc} is the correlated range error standard deviation; D_x , D_y , and D_z are the dilutions of precision resolved along the x , y , and z axes of the ECI or ECEF frame; and D_T is the TDOP (see Section 7.1.4).

The corrected curvilinear position uncertainties are given by

$$\begin{pmatrix} \sigma_L \\ \sigma_\lambda \\ \sigma_b \\ \sigma_{prc} \end{pmatrix} = \begin{pmatrix} \sqrt{P_{1,1} + \Delta \sigma_L^2} \\ \sqrt{P_{2,2} + \Delta \sigma_\lambda^2} \\ \sqrt{P_{3,3} + \Delta \sigma_b^2} \\ \sqrt{P_{7,7} + \Delta \sigma_{prc}^2} \end{pmatrix} \quad (7.208)$$

where

$$\begin{aligned}\Delta\sigma_L &= \frac{D_N\sigma_{\rho c}}{R_N(L_a) + h_a}, \quad \Delta\sigma_\lambda = \frac{D_E\sigma_{\rho c}}{[R_E(L_a) + h_a] \cos L_a} \\ \Delta\sigma_b &= D_D\sigma_{\rho c}, \quad \Delta\sigma_{\rho rc} = D_T\sigma_{\rho c}\end{aligned}\quad (7.209)$$

and D_N , D_E , and D_D are given by (7.55).

Where there is a change in the satellites tracked by the receiver, known as a constellation change, or there is an ephemeris update, the error in the navigation solution due to the correlated range errors will change. The Kalman filter will respond more quickly to this change if the position and clock-offset state uncertainties are boosted. Where the range biases are estimated as states, the relevant state should instead be reset on a constellation change or ephemeris update.

7.5.3 Combined Navigation and Tracking

In conventional GNSS user equipment, the information from the baseband signal processing channels, the Is and Qs, is filtered by the code and carrier tracking loops before being passed to the navigation processor. This smoothes out noise and enables the navigation processor to be iterated at a lower rate. However, it also filters out some of the signal information. Each set of pseudo-range and pseudo-range rate measurements input to the navigation processor is derived from several successive sets of Is and Qs, but the older data has been down-weighted by the tracking loops, partially discarding it.

Once an initial navigation solution has been obtained, the signal tracking and navigation-solution determination can be combined into a single estimation algorithm, usually Kalman filter-based. Figure 7.31 illustrates this. The Is and Qs are used to directly estimate corrections to the navigation solution, from which

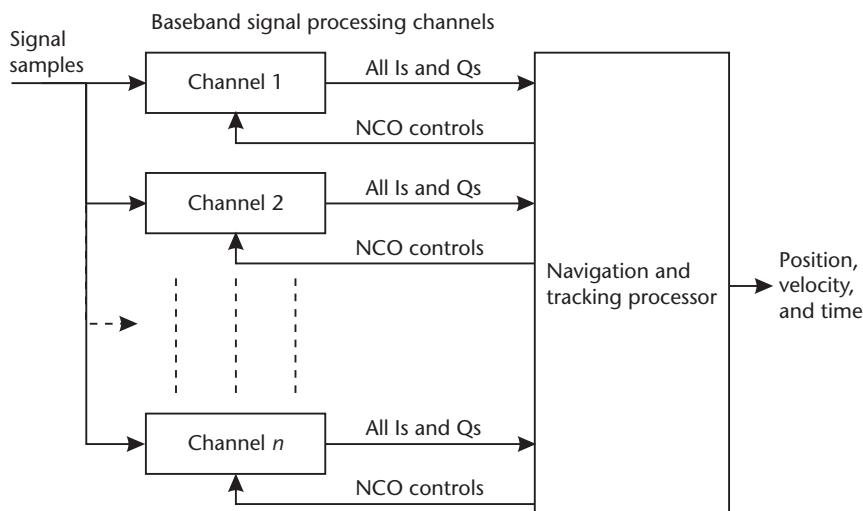


Figure 7.31 Combined navigation and tracking loop.

the NCO control commands are derived. This is sometimes known as vector tracking. It brings the benefit that all I and Q data is weighted equally in the navigation solution, reducing the impact of tracking errors. Where the navigation solution is overdetermined, the tracking of each signal is aided by the others and tracking lock may be maintained through a single channel outage. Thus, a given navigation-solution precision can be obtained in a poorer signal-to-noise environment. The main drawback is an increase in processor load. When fewer than four satellites are tracked, the user equipment must revert to conventional tracking.

The simplest implementation of combined navigation and tracking is the vector delay lock loop [18]. This uses the navigation processor to track code, but retains independent carrier tracking loops. Where four GNSS signals are tracked, the states and system model are the same as for the conventional navigation filter described in the previous section. Discriminator functions are used to obtain a measurement of each code tracking error, $\tilde{x}_{j,k}$, from the Is and Qs as described in Section 7.3.2. Using (7.68), each pseudo-range measurement innovation may then be obtained:

$$\begin{aligned}\delta z_{pj,k}^- &= \rho_{Cj,k}^- - \hat{\rho}_{Cj,k}^- + w_{mpj,k} \\ &\approx \rho_{Rj,k}^- - \hat{\rho}_{Rj,k}^- + w_{mpj,k} \\ &= -\frac{c}{f_{co}} \tilde{x}_{j,k}\end{aligned}\quad (7.210)$$

where w_m is the measurement noise. The pseudo-range-rate measurements are obtained from the carrier tracking loops and perform the same role as the range-rate aiding inputs to conventional code tracking loops. The measurement model described in Section 7.5.2.3 may then be applied.

Where signals on more than one frequency are tracked, the ionosphere propagation delay for each satellite must be estimated as a Kalman filter state. Where more than four satellites are tracked, range-bias estimation can be used to maintain code tracking at the peak of each signal's correlation function. This is more important where the correlation peak is narrow, as applies to many BOC signals, and/or only one frequency is used, in which case the range biases are larger.

Each corrected pseudo-range, $\hat{\rho}_{Cj}$, may be calculated from the Kalman filter estimates and the satellite position and velocity data, in the simplest case using (7.199). Estimates of each raw pseudo-range, $\hat{\rho}_{Rj}$, are then obtained by subtracting the satellite clock, ionosphere, and troposphere corrections, as given by (7.43). By analogy with (7.97), the code NCO command is then

$$\hat{f}_{co,NCOj,k+1} = f_{co} \left[1 - \frac{(\hat{\rho}_{Rj,k+1}^- - \hat{\rho}_{Rj,k}^-)}{c\tau} \right] \quad (7.211)$$

where τ is the Kalman filter update interval, which can be longer than the correlation period, τ_a , with $\tilde{x}_{j,k}$ averaged over τ . However, the update rate must be at least twice the code tracking bandwidth, B_{LCO} , which is not constant in a Kalman filter implementation.

The vector DLL may be extended to incorporate carrier frequency tracking by deriving each pseudo-range-rate measurement innovation from a carrier-frequency discriminator (see Section 7.2.3):

$$\begin{aligned}\delta z_{rj, k}^- &= \dot{\rho}_{Cj, k} - \hat{\dot{\rho}}_{Cj, k}^- + w_{mrj, k} \\ &\approx \dot{\rho}_{Rj, k} - \hat{\dot{\rho}}_{Rj, k}^- + w_{mrj, k} \\ &= -\frac{c}{f_{co}} \tilde{\delta f}_{caj, k}\end{aligned}\quad (7.212)$$

The carrier NCO command is then

$$\hat{f}_{ca, \text{NCO}j, k+1} = f_{IF} - \frac{f_{ca}}{c} \hat{\dot{\rho}}_{Rj, k+1}^- \quad (7.213)$$

No additional Kalman filter states are essential for carrier frequency tracking, but the addition of acceleration and rate-of-clock-drift states enables second-order, as opposed to first-order, frequency tracking.

To implement carrier phase tracking with a combined navigation and tracking algorithm, the pseudo-range rate measurement inputs to the Kalman filter are replaced by carrier phase measurements. The iteration rate must also be increased to accommodate the carrier-phase tracking bandwidth, which is typically higher than the code tracking or carrier-frequency-tracking bandwidth.

Instead of separate code and carrier phase measurements, the Kalman filter may input the Is and Qs as its measurements directly [57]. The measurement matrix, H_G , is obtained by differentiating (7.66). This is a form of coherent tracking, as the carrier phase must be known in order to track the code. Coherent tracking is less noisy and, in vector form, avoids discriminator nonlinearities. However, when there is insufficient signal to noise to track carrier phase, the code cannot be tracked either.

If the carrier phase error of each signal tracked is estimated as a Kalman filter state, carrier phase can be tracked without keeping the reference carrier phase within the receiver synchronized with that of the signal because the phase difference is then known. A measurement innovation may be formed from the difference between the carrier phase discriminator output and the value predicted from the Kalman filter estimates, noting that in conventional tracking, the predicted discriminator output is always zero. Consequently, the rate at which the NCO commands are updated (but not the measurement update role) can be less than twice the carrier-phase tracking bandwidth. Similarly, a greater lag between generating the NCO commands and receiving the corresponding Is and Qs may be tolerated.

However, the receiver's reference carrier frequency and code do need to be kept synchronized with the signal; otherwise, the two do not correlate and the signal is absent from the I and Q measurements. Thus, the NCO command update rate and control lag must be designed to accommodate the code tracking and carrier-frequency tracking bandwidths necessary to respond to the user dynamics and the noise on the receiver's reference oscillator.

Adding carrier-phase states to the Kalman filter and iterating it at a higher rate significantly increases the processor load. A common solution is to use a bank of prefilters to track the carrier phase at a higher rate than the main navigation and tracking algorithm [58].

As in a conventional system, the Is and Qs should also be used to measure the carrier power to noise density. If the c/n_0 measurements are used to determine the Kalman filter's measurement noise covariance matrix, \mathbf{R}_G , the measurements will be optimally weighted and the state uncertainties (obtained from the error covariance matrix, \mathbf{P}) may be used to determine code tracking lock.

7.5.4 Position Error Budget

The error in a GNSS position solution is determined by the range errors, described in Section 7.4, and the signal geometry, discussed in Section 7.1.4. Every published GNSS error budget is different, making varying assumptions about the system performance, receiver design, number of satellites tracked, mask angle, and multipath environment.

The values presented here assume the 2005 post-L-AII GPS satellite clock and ephemeris errors [39]. BPSK(1), BOC_s(1,1), BPSK(10), and BOC_s(10,5) signal modulations are considered. Tracking noise is calculated assuming $C/N_0 = 40$ dB-Hz, $\tau_a = 20$ ms, a dot-product power discriminator is used, $B_{L_CO} = 1$ Hz, and the receiver precorrelation bandwidth matches the transmission bandwidth. Three multipath environments are considered, each with a factor of 5 attenuation of the delayed signal and a uniform delay distribution of 0–2m for the short-range model, 0–20m for the medium-range model, and 1–200m for the long-range model. Table 7.6 lists the standard deviations of the range error components.

Taking a weighted average of the DOP values in Table 7.4, assuming a 24-satellite GPS constellation with all satellites in view tracked, the PDOP is 1.67, HDOP is 0.88, and VDOP is 1.42. Table 7.7 presents the position error budget.

Table 7.6 Contributions to the Range Error Standard Deviation

Source	Range Error Standard Deviation (m)
Residual satellite clock error	1.0
Ephemeris error	0.45
Residual ionosphere error (single-frequency user)	4.0
Residual ionosphere error (dual-frequency user)	0.1
Residual troposphere error	0.2
Tracking noise for:	
BPSK(1) signal, $d = 0.1$	0.67
BOC _s (1,1) signal, $d = 0.1$	0.39
BPSK(10) signal, $d = 1$	0.21
BOC _s (10,5) signal	0.06
Short-range multipath error	0.1
Medium-range multipath error	0.94
Long-range multipath error for:	
BPSK(1) signal, $d = 0.1$	1.44
BOC _s (1,1) signal, $d = 0.1$	1.33
BPSK(10) signal, $d = 1$	0.12
BOC _s (10,5) signal [14]	0.23

Table 7.7 Position Error Budget

Frequencies	Multipath	Signal	Position Error Standard Deviation (m)		
			Total	Horizontal (Radial)	Vertical
Single	Short-range	BPSK(1)	7.0	3.7	6.0
		BOC _s (1,1)	6.9	3.7	5.9
		BPSK(10)	6.9	3.7	5.9
		BOC _s (10,5)	6.9	3.7	5.9
	Medium-range	BPSK(1)	7.2	3.8	6.1
		BOC _s (1,1)	7.1	3.7	6.0
		BPSK(10)	7.1	3.7	6.0
		BOC _s (10,5)	7.1	3.7	6.0
	Long-range	BPSK(1)	7.4	3.9	6.3
		BOC _s (1,1)	7.3	3.9	6.2
		BPSK(10)	6.9	3.7	5.9
		BOC _s (10,5)	6.9	3.7	5.9
Dual	Short-range	BPSK(1)	2.2	1.2	1.9
		BOC _s (1,1)	2.0	1.0	1.6
		BPSK(10)	2.0	1.0	1.6
		BOC _s (10,5)	2.0	1.0	1.6
	Medium-range	BPSK(1)	2.7	1.4	2.3
		BOC _s (1,1)	2.5	1.3	2.1
		BPSK(10)	2.5	1.3	2.1
		BOC _s (10,5)	2.5	1.3	2.1
	Long-range	BPSK(1)	3.2	1.7	2.8
		BOC _s (1,1)	3.0	1.6	2.5
		BPSK(10)	1.9	1.0	1.6
		BOC _s (10,5)	1.9	1.0	1.6

References

- [1] Van Dierendonck, A. J., “GPS Receivers,” in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 329–407.
- [2] Dorsey, A. J., et al., “GPS System Segments,” in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 67–112.
- [3] Misra, P., and P. Enge, *Global Positioning System Signals, Measurements, and Performance*, Lincoln, MA: Ganga-Jamuna Press, 2001.
- [4] Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, New York: Wiley, 2001.
- [5] Anon., *Navstar GPS Space Segment/Navigation User Interfaces*, IS-GPS-200, Revision D, ARINC, December 2004.
- [6] Kaplan, E. D., et al., “Fundamentals of Satellite Navigation,” in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 21–65.
- [7] Ward, P. W., J. W. Betz, and C. J. Hegarty, “GPS Satellite Signal Characteristics,” in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 113–151.
- [8] Di Esposti, R., “Time-Dependency and Coordinate System Issues in GPS Measurement Models,” *Proc. ION GPS 2000*, Salt Lake City, UT, September 2000, pp. 1925–1929.
- [9] Conley, R., et al., “Performance of Stand-Alone GPS,” in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 301–378.

- [10] Kraus, J., and R. Marhefka, *Antennas*, 3rd ed., New York: McGraw-Hill, 2001.
- [11] Johnson, R. C., and H. Jasik, *Antenna Engineering Handbook*, 3rd ed., New York: McGraw-Hill, 1993.
- [12] Vittorini, L. D., and B. Robinson, "Receiver Frequency Standards: Optimizing Indoor GPS Performance," *GPS World*, November 2003, pp. 40–48.
- [13] Pratt, A. R., "g-Effects on Oscillator Performance in GPS Receivers," *Navigation: JION*, Vol. 36, No. 1, 1989, pp. 63–75.
- [14] Ward, P. W., J. W. Betz, and C. J. Hegarty, "Satellite Signal Acquisition, Tracking and Data Demodulation," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 153–241.
- [15] Bao-Yen Tsui, J., *Fundamentals of Global Positioning System Receivers: A Software Approach*, 2nd ed., New York: Wiley, 2004.
- [16] Hegarty, C., et al., "Suppression of Pulsed Interference Through Blanking," *Proc. ION 56th AM*, San Diego, CA, June 2000, pp. 399–408.
- [17] Akos, D. M., et al., "Real-Time GPS Software Radio Receiver," *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 809–816.
- [18] Spilker, J. J., Jr., "Fundamentals of Signal Tracking Theory," in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 245–327.
- [19] Groves, P. D., "GPS Signal to Noise Measurement in Weak Signal and High Interference Environments," *Navigation: JION*, Vol. 52, No. 2, 2005, pp. 83–92.
- [20] Hein, G. W., et al., "Performance of Galileo L1 Signal Candidates," *Proc. ENC GNSS 2004*, Rotterdam, Netherlands, May 2004.
- [21] Ward, P. W., J. W. Betz, and C. J. Hegarty, "Interference, Multipath and Scintillation," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 243–299.
- [22] Betz, J. W., and K. R. Kolodziejski, "Extended Theory of Early-Late Code Tracking for a Bandlimited GPS Receiver," *Navigation: JION*, Vol. 47, No. 3, 2000, pp. 211–226.
- [23] Tran, M., and C. Hegarty, "Receiver Algorithms for the New Civil GPS Signals," *Proc. ION NTM*, San Diego, CA, January 2002, pp. 778–789.
- [24] Dafesh, P., et al., "Description and Analysis of Time-Multiplexed M-Code Data," *Proc. ION 58th AM*, Albuquerque, NM, June 2002, pp. 598–611.
- [25] Harrison, D., et al., "A Fast Low-Energy Acquisition Technology for GPS Receivers," *Proc. ION 55th AM*, Cambridge, MA, June 1999, pp. 433–441.
- [26] Lee, W. C., et al., "Fast, Low Energy GPS Navigation with Massively Parallel Correlator Array Technology," *Proc. ION 55th AM*, Cambridge, MA, June 1999, pp. 443–450.
- [27] Scott, L., A. Jovancevic, and S. Ganguly, "Rapid Signal Acquisition Techniques for Civilian & Military User Equipment Using DSP Based FFT Processing," *Proc. ION GPS 2001*, Salt Lake City, UT, September 2001, pp. 2418–2427.
- [28] Lin, D. M., and J. B. Y. Tsui, "An Efficient Weak Signal Acquisition Algorithm for a Software GPS Receiver," *Proc. ION GPS 2001*, Salt Lake City, UT, September 2001, pp. 115–119.
- [29] Ziedan, N. I., *GNSS Receivers for Weak Signals*, Norwood, MA: Artech House, 2006.
- [30] Betz, J. W., "Design and Performance of Code Tracking for the GPS M Code Signal," *Proc. ION GPS 2000*, Salt Lake City, UT, September 2000, pp. 2140–2150.
- [31] Betz, J. W., "Binary Offset Carrier Modulation for Radionavigation," *Navigation: JION*, Vol. 48, No. 4, 2001, pp. 227–246.
- [32] Fine, P., and W. Wilson, "Tracking Algorithm for GPS Offset Carrier Signals," *Proc. ION NTM*, San Diego, CA, January 1999, pp. 671–676.
- [33] Dovis, F., P. Mulassano, and L. Lo Presti, "A Novel Algorithm for the Code Tracking of BOC(n,n) Modulated Signals," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 152–155.

- [34] Ward, P., "Performance Comparisons Between FLL, PLL and a Novel FLL-Assisted PLL Carrier Tracking Loop Under RF Interference Conditions," *Proc. ION GPS-98*, Nashville, TN, September 1998, pp. 783–795.
- [35] Betz, J. W., "Effect of Partial-Band Interference on Receiver Estimation of C/N0: Theory," *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 817–828.
- [36] Ross, J. T., J. L. Leva, and S. Yoder, "Effect of Partial-Band Interference on Receiver Estimation of C/N0: Measurements," *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 829–838.
- [37] Hwang, P. Y., G. A. McGraw, and J. R. Bader, "Enhanced Differential GPS Carrier-Smoothed Code Processing Using Dual-Frequency Measurements," *Navigation: JION*, Vol. 46, No. 2, 1999, pp. 127–137.
- [38] Ashby, N., and J. J. Spilker, Jr., "Introduction to Relativistic Effects on the Global Positioning System," in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 623–697.
- [39] Creel, T., et al., "New, Improved GPS: The Legacy Accuracy Improvement Initiative," *GPS World*, March 2006, pp. 20–31.
- [40] Olynik, M., et al., "Temporal Variability of GPS Error Sources and Their Effect on Relative Position Accuracy," *Proc. ION NTM*, San Diego, CA, January 2002, pp. 877–888.
- [41] Klobuchar, J. A., "Ionosphere Effects on GPS," in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 485–515.
- [42] Groves, P. D., and S. J. Harding, "Ionosphere Propagation Error Correction for Galileo," *Journal of Navigation*, Vol. 56, No. 1, 2003, pp. 45–50.
- [43] Radicella, S. M., and R. Leitinger, "The Evolution of the DGR Approach to Model Electron Density Profiles," *Advances in Space Research*, Vol. 27, No. 1, 2001, pp. 35–40.
- [44] Spilker, J. J., Jr., "Tropospheric Effects on GPS," in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 517–546.
- [45] Mendes, V. B., and R. B. Langley, "Tropospheric Zenith Delay Prediction Accuracy for High-Precision GPS Positioning and Navigation," *Navigation: JION*, Vol. 46, No. 1, 1999, pp. 25–34.
- [46] Powe, M., J. Butcher, and J. Owen, "Tropospheric Delay Modelling and Correction Dissemination Using Numerical Weather Prediction Fields," *Proc. GNSS 2003*, {CENCF2}, Graz, Austria, April 2003.
- [47] Jupp, A., et al., "Use of Numerical Weather Prediction Fields for the Improvement of Tropospheric Corrections in Global Positioning Applications," *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 377–389.
- [48] Van Dierendonck, A. J., P. Fenton, and T. Ford, "Theory and Performance of a Narrow Correlator Spacing in a GPS Receiver," *Navigation: JION*, Vol. 39, No. 3, 1992, pp. 265–283.
- [49] Ries, L., et al., "Tracking and Multipath Performance Assessments of BOC Signals Using a Bit-Level Signal Processing Simulator," *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 1996–2010.
- [50] Braasch, M. S., "Multipath Effects," in *Global Positioning System: Theory and Applications Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 547–568.
- [51] Van Nee, R. D. J., "GPS Multipath and Satellite Interference," *Proc. ION 48th AM*, Washington, D.C., June 1992, pp. 167–177.
- [52] Irsigler, M., and B. Eissfeller, "Comparison of Multipath Mitigation Techniques with Consideration of Future Signal Structures," *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 2584–2592.

- [53] Braasch, M. S., "Autocorrelation Sidelobe Considerations in the Characterization of Multipath Errors," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 33, No. 1, 1997, pp. 290–295.
- [54] Hegarty, C. J., "Least-Squares and Weighted Least-Squares Estimates," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 663–669.
- [55] Farrell, J. A., and M. Barth, *The Global Positioning System and Inertial Navigation*, New York: McGraw Hill, 1999.
- [56] Axelrad, P., and R. G. Brown, "Navigation Algorithms," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 409–493.
- [57] Sennott, J. W., and D. Senffner, "Navigation Receiver with Coupled Signal-Tracking Channels," U.S. Patent 5,343,209, 1994.
- [58] Abbott, A. S., and W. E. Lillo, "Global Positioning System and Inertial Measuring Unit Ultra-Tight Coupling Method," U.S. Patent 6,916,025, 2003.

Selected Bibliography

- Braasch, M. S., and A. J. Van Dierendonck, "GPS Receiver Architectures and Measurements," *Proc. IEEE*, Vol. 87, No. 1, January 1999, pp. 48–64.
- Holmes, J. K., *Spread Spectrum Systems for GNSS and Wireless Communications*, Norwood, MA: Artech House, 2007.
- Van Dierendonck, A. J., "Satellite Radio Navigation," in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 178–282.

Endnotes

1. This and subsequent paragraphs are based on material written by the author for QinetiQ, so comprise QinetiQ copyright material.
2. End of QinetiQ copyright material.
3. This paragraph, up to this point, is based on material written by the author for QinetiQ, so comprises QinetiQ copyright material.

Advanced Satellite Navigation

The preceding chapters described the satellite navigation systems and their user equipment. This chapter reviews a number of techniques that enhance the accuracy, robustness, and reliability of GNSS. Section 8.1 discusses how additional infrastructure may be used to improve GNSS positioning accuracy using differential techniques, while Section 8.2 describes how carrier phase techniques may be used to obtain high-precision position and attitude measurements under good conditions.

Sections 8.3 and 8.4 review techniques for improving GNSS robustness under difficult conditions, covering operation in poor signal-to-noise environments and multipath mitigation, respectively. Section 8.5 describes how signal monitoring networks may be used to protect users from the effects of faulty satellite signals. Finally, Section 8.6 describes how SPS GPS users can use semi-codeless tracking to gain the benefit of dual-frequency operation with legacy signals.

8.1 Differential GNSS

The correlated range errors due to ephemeris prediction errors and residual satellite clock, ionosphere, and troposphere errors vary slowly with time and user location. Therefore, by comparing pseudo-range measurements with those made by equipment at a presurveyed location, known as a *reference station* or base station, the correlated range errors may be calibrated out, improving the navigation-solution accuracy. This is the principle behind DGNSS. Figure 8.1 illustrates the concept.

This section describes some different implementations of DGNSS, covering a local area with a single reference station or a regional or wide area with multiple reference stations. Before this, the spatial and temporal correlation properties of the various GNSS error sources are discussed, while the section concludes with a description of relative GNSS.

8.1.1 Spatial and Temporal Correlation of GNSS Errors

Table 8.1 gives typical values for the variation of correlated GNSS error sources with time and space [1, 2]. This gives an indication of how the accuracy of the DGNSS navigation solution varies with the separation of the user from the reference station and the latency of the calibration data. The divergence in correlated range errors as the user moves away from a reference station is known as *spatial decorrelation*, while the divergence due to differences in measurement time is known as *time decorrelation*.

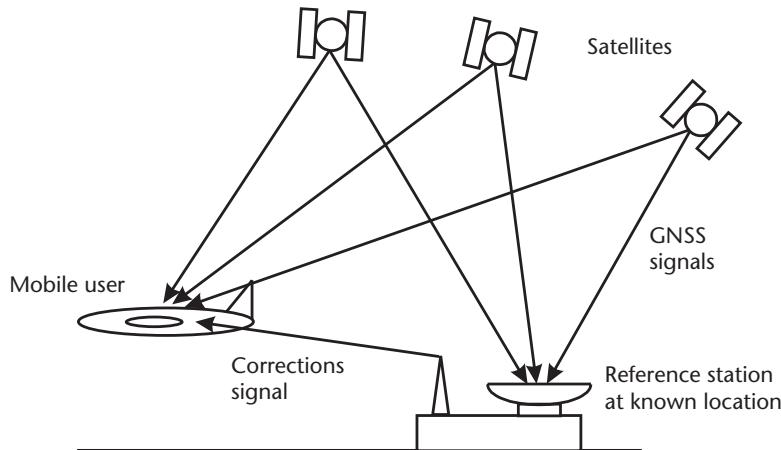


Figure 8.1 Schematic of differential GNSS.

Table 8.1 Typical Variation of Correlated GNSS Error Sources over Time and Space

Error Source	Variation over 100s	Variation over 100 km	Variation over 1 km
		Horizontal Separation	Vertical Separation
Residual satellite clock	~0.1m	None	None
Ephemeris	~0.02m	~0.01m	Negligible
Ionosphere (uncorrected)	0.1–0.4m	0.2–0.5m	Negligible
Troposphere (uncorrected)	0.1–1.5m	0.1–1.5m	1–2m*

*Ground reference station

The satellite clock errors are the same for all observers, while the spatial variation of the ephemeris errors is very small. Following the L-AII, the temporal variation of these errors for GPS is also small. The variation in ionosphere and troposphere propagation errors is much greater and depends on the elevation angle, time of day, and weather.

The tracking and multipath errors are uncorrelated between users at different locations, so cannot be corrected using DGNSS. Therefore, these errors must be minimized in the reference station to prevent them from disrupting the mobile user's navigation solution. A narrow early-late correlator spacing and narrow tracking-loop bandwidths (see Section 7.4.3), combined with carrier-smoothing of the pseudo-range measurements (Section 7.3.7) and use of a high-performance reference oscillator (Section 7.2.2), minimizes the tracking errors. The narrow correlator spacing also reduces the impact of multipath interference (Section 7.4.4), while further multipath mitigation techniques are discussed in Section 8.4.

8.1.2 Local and Regional Area DGNSS

In a local area DGNSS (LADGNSS) system, corrections are transmitted from a single reference station to mobile users, sometimes known as rovers, within the range of its transmitter. The closer the user is to the reference station, the more accurate the navigation solution is. Users within a 150-km horizontal radius typically achieve an accuracy of about 1m.

Transmitting corrections to the position solution requires all users and the reference station to use the same set of satellites so that the correlated errors affecting each satellite are cancelled out. This is not practical, as satellite signals are intermittently blocked by buildings, terrain, and sometimes the host vehicle body. Instead, range corrections are transmitted, allowing the user to select any combination of the satellites tracked by the reference station. The corrections may be subject to reference-station receiver clock errors. However, this does not present a problem, as the user's navigation processor simply solves for the relative clock offset and drift between the user and reference instead of the user receiver clock errors.

To obtain differentially corrected pseudo-range measurements, $\tilde{\rho}_{DCj}$, differential corrections, $\Delta\rho_{dcj}$, may be applied either in place of the satellite clock, ionosphere, and troposphere corrections:

$$\tilde{\rho}_{DCj} = \tilde{\rho}_{Rj} + \Delta\rho_{dcj} \quad (8.1)$$

or in addition to these corrections:

$$\tilde{\rho}_{DCj} = \tilde{\rho}_{Rj} + \Delta\rho_{icj} + \Delta\rho_{tcj} + \Delta\rho_{scj} + \Delta\rho_{dcj} \quad (8.2)$$

where the notation is as defined in Section 7.1.2. Application of only some of these corrections is also valid. However, it is essential that the same convention is adopted by both the reference station and the users. The ionosphere correction obtained from dual-frequency measurements is generally more accurate than that from DGNSS, while a troposphere model should be used for air applications as the troposphere errors vary significantly with height.

Most LADGNSS systems adopt the Radio Technical Committee for Maritime Services (RTCM) Special Committee (SC) 104 transmission protocol. This supports a number of different messages, enabling each LADGNSS system's transmissions to be tailored to the user base and data rate, which can be as low as 50 bit s⁻¹ [3, 4]. Range-rate corrections are transmitted to enable users to compensate for latency in the range corrections, while *delta corrections* are transmitted for users of old ephemeris and clock data broadcast by the satellite constellation.

Many LADGNSS stations transmit in the 283.5–325-kHz marine radio-beacon band, with coverage radii of up to 500 km. VHF and UHF band data links, cell phone systems, radio and television broadcasts, the Internet, and Loran signals (Section 9.2.2) are also used.

Regional area DGNSS (RADGNSS) enables LADGNSS users to obtain greater accuracy by using corrections from multiple reference stations, combined using

$$\Delta\rho_{dcj} = \sum_i W_i \Delta\rho_{dcj,i}, \quad \sum_i W_i = 1 \quad (8.3)$$

where the weighting factors, W_i , are determined by the user's distance from each reference station. RADGNSS may be implemented entirely within the receiver, or corrections from multiple reference stations may be included in a single transmission.

8.1.3 Wide Area DGNSS

A wide area DGNSS (WADGNSS) system aims to provide positioning to meter accuracy over a continent, such as Europe, or large country, such as the United States, using much fewer reference stations than LADGNSS or RADGNSS would require. It does this by operating on the same principle as the GNSS control segments described in Chapter 6. Typically, 10 or more reference stations at known locations send pseudo-range and dual-frequency ionosphere delay measurements to an MCS. The MCS then computes corrections to the GNSS system broadcast ephemeris and satellite clock parameters, together with a grid of ionosphere model coefficients, which are transmitted to the users [5–7].

WADGNSS is one of the functions of the SBAS systems, described in Section 6.2.4, and is also intended to form part of the Galileo commercial service (see Section 6.4). Other satellite-delivered WADGNSS services include NASA's Global Differential GPS System [8] and the commercial OmniStar [9] and StarFire [10] systems. WADGNSS data can also be transmitted to users via terrestrial radio links, cell phones, and the Internet.

Using a network of reference stations denser than the system control segments enables WADGNSS to achieve more accurate ephemeris and satellite clock calibration over the area spanned by the reference stations, while the ionosphere model coefficients are optimized over the service area, rather than globally. However, improvements in the accuracy of the ephemeris and satellite clock data broadcast by the GNSS satellites, together with the advent of dual-frequency ionosphere correction for civil users, could limit the benefits of WADGNSS.

8.1.4 Precise Point Positioning

Precise point positioning (PPP) obtains decimeter accuracy from stand-alone GNSS user equipment by calibrating each of the main error sources separately [9, 11]. Code tracking errors are reduced by a combination of carrier smoothing and time averaging, though an initialization period of about 20 minutes is needed to achieve centimeter accuracy with GPS C/A code. Ionosphere delays are calibrated with dual-frequency measurements, while residual troposphere propagation errors can be reduced with water vapor measurement and selection of higher elevation satellites.

The ephemeris and satellite clock errors are reduced using precision orbit and clock products in place of the navigation message data. The International GNSS Service (IGS) is a voluntary network of more than 200 organizations in more than 80 countries, operating more than 370 reference stations [12]. Their freely available postprocessed orbit and clock products are accurate to within 5 cm, making PPP a useful technique for surveying.

For navigation, real-time ephemeris and satellite clock data is required. IGS predicted clock data is not much better than that broadcast by the GNSS satellites. However, the OmniStar High Performance (HP) service and StarFire provide real-time decimeter accuracy positioning after a 20-minute initialization [13, 14].

8.1.5 Relative GNSS

Relative GNSS (RGNSS) is used in applications, such as shipboard landing of aircraft and in-flight refueling, where the user position must be known accurately with respect to the reference station, but the position accuracy with respect to the Earth is less important. This relative position is known as a *baseline*. In RGNSS, the reference station transmits absolute pseudo-range measurements, which are then differenced with the user's pseudo-range measurements:

$$\Delta \tilde{\rho}_{Rj} = \tilde{\rho}_{Rj}(u) - \tilde{\rho}_{Rj}(r) \quad (8.4)$$

where the suffixes u and r denote the user and reference station, respectively. Then, from (7.43) and (7.172), assuming the user and reference are close enough for the ionosphere, troposphere, and Sagnac corrections to cancel,

$$\Delta \tilde{\rho}_{Rj} \approx \left| \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ea}^e(t_{sa}) \right| - \left| \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{er}^e(t_{sa}) \right| + \Delta \hat{\rho}_{rc}^e(t_{sa}) + \delta \hat{\rho}_{\epsilon j}^+ \quad (8.5)$$

where the subscript r denotes the reference station body frame, and $\Delta \hat{\rho}_{rc}$ is the relative receiver clock error. By analogy with (7.184), the least-squares relative position solution is then

$$\begin{pmatrix} \hat{\mathbf{r}}_{ra}^e(t_{sa}) \\ \Delta \hat{\rho}_{rc}^e(t_{sa}) \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{r}}_{ra}^e(t_{sa}) - \hat{\mathbf{r}}_{er}^e(t_{sa}) \\ \Delta \hat{\rho}_{rc}^e(t_{sa}) \end{pmatrix} = (\mathbf{G}_e^T \mathbf{G}_e)^{-1} \mathbf{G}_e^{-1} \begin{pmatrix} \Delta \tilde{\rho}_{R1} \\ \Delta \tilde{\rho}_{R2} \\ \vdots \\ \Delta \tilde{\rho}_{Rn} \end{pmatrix} \quad (8.6)$$

8.2 Carrier-Phase Positioning and Attitude

Where the user and reference station are relatively close, the accuracy of code-based differential GNSS is determined by the tracking and multipath errors. However, in range terms, carrier phase tracking is much less noisy and exhibits smaller multipath errors than code tracking. Therefore, by performing relative positioning with carrier measurements as well as code measurements, centimeter accuracy is attainable.

The range from satellite j to the user, subject to ionosphere and troposphere propagation errors, is

$$\rho_j(t_{sa}) = \left(\frac{\phi'_{ca,j}(t_{sa}) - \phi'_{s,j}(t_{st})}{2\pi} + a_j(t_{sa}) \right) \lambda_{ca,j} \quad (8.7)$$

where $\phi'_{ca,j}$ is the carrier phase offset at the user equipment antenna, $\phi'_{s,j}$ is the carrier phase offset at the satellite antenna, a_j is an integer number of carrier cycles (many authors use N) and $\lambda_{ca,j}$ is the carrier wavelength, compensated for the Doppler shift. The carrier phase offset at the satellite is unknown, so must be cancelled out by reference-station measurements alongside the ionosphere and

troposphere errors. The receiver's carrier phase measurements are also subject to an offset, common to all tracking channels. This must be solved as part of the navigation-solution determination, like the receiver clock offset. The measured carrier-phase pseudo-range is thus

$$\tilde{\rho}_{Rca,j}(t_{sa}) = \left(\frac{\tilde{\phi}'_{ca,j}(t_{sa})}{2\pi} + a_j(t_{sa}) \right) \lambda_{ca,j} \quad (8.8)$$

where, with reference to (7.62),

$$\tilde{\phi}'_{ca,j}(t_{sa}) = [2\pi\Delta\tilde{f}_{ca,j}t_{sa} + \tilde{\phi}_{ca,j}(t_{sa})] \text{MOD}2\pi \quad (8.9)$$

where the MOD operator gives the remainder from integer division.

The number of candidate values of a_j is constrained by the code pseudo-range, however there are still a number of values that must be selected from; hence, a_j is known as the *integer ambiguity*. Where a Costas carrier-phase discriminator is used, due to the presence of navigation data bits, the carrier phase measurement can be half a cycle out. The user equipment can correct for this using the sign of known bits in the navigation data message. Where this is not done, a_j can take half-integer values as well as integer values. Ambiguity resolution is discussed in Section 8.2.1. However, using carrier phase measurements alone, the integer ambiguity changes between successive sets of measurements. This presents a problem, as ambiguity resolution algorithms generally require more than one set of measurements. If the accumulated delta range is used, the integer ambiguity remains constant:

$$\tilde{\rho}_{Rca,j}(t_{sa}) = \Delta\tilde{\rho}_{ADR,j}(t_{sa}) + \left(\frac{\tilde{\phi}'_{ca,j}(t_{0,j})}{2\pi} + a_j \right) \lambda_{ca,j} \quad (8.10)$$

where $t_{0,j}$ is the carrier tracking initialization time for channel j . However, the carrier phase measurement at initialization is then required. In many GNSS user equipment designs, this is simply subtracted from the ADR measurement. The modified ADRs may be obtained directly by combining a current carrier phase measurement with a count of the integer change in carrier cycles since carrier tracking initialization [15]. Thus,

$$\Delta\tilde{\rho}'_{ADR,j}(t_{sa}) = \left(\frac{\tilde{\phi}'_{ca,j}(t_{sa})}{2\pi} + \tilde{a}_j(t_{sa}) - \tilde{a}_j(t_{s0,j}) \right) \lambda_{ca,j} \quad (8.11)$$

enabling (8.10) to be simplified to

$$\tilde{\rho}_{Rca,j}(t_{sa}) = \Delta\tilde{\rho}'_{ADR,j}(t_{sa}) + a_j(t_{s0,j}) \lambda_{ca,j} \quad (8.12)$$

Relative carrier phase measurements between the user (u) and reference station (r) are formed using

$$\Delta\tilde{\rho}_{Rca,j}(t_{sa}) = \Delta\tilde{\rho}'_{ADR,j}(t_{sa}, u) - \Delta\tilde{\rho}'_{ADR,j}(t_{sa}, r) \quad (8.13)$$

By analogy with (8.5),

$$\Delta \tilde{\rho}_{Rj} \approx \left| \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{ea}^e(t_{sa}) \right| - \left| \hat{\mathbf{r}}_{esj}^e(\tilde{t}_{st,j}) - \hat{\mathbf{r}}_{er}^e(t_{sa}) \right| + \Delta \hat{\rho}_{r\phi}(t_{sa}) + \Delta a_j + \delta \rho_{\epsilon j}^+ \quad (8.14)$$

where Δa_j is the relative integer ambiguity and $\Delta \hat{\rho}_{r\phi}$ is the relative receiver phase offset, which also absorbs small time-synchronization errors between the user and reference station. Once the relative integer ambiguities are resolved, the baseline, $\hat{\mathbf{r}}_{ra}^e$, and receiver phase offset may be solved using (8.6). However, it is common practice to form double-difference measurements of the form

$$\nabla \Delta \tilde{\rho}_{Rca,ij}(t_{sa}) = \Delta \tilde{\rho}_{Rca,i}(t_{sa}) - \Delta \tilde{\rho}_{Rca,j}(t_{sa}) \quad (8.15)$$

by differencing relative carrier phase measurements for different satellites. This cancels out $\Delta \hat{\rho}_{r\phi}$, but also reduces by one the number of relative integer ambiguities to find in order to determine a relative navigation solution.

Because GNSS signals are circularly polarized, rotations of the antenna change the measured carrier phase by one cycle per complete antenna rotation. This is known as *phase wind-up* and may be compensated where an attitude solution is available.

Carrier-phase positioning techniques that can operate in real-time over moving baselines are known as real-time kinematic (RTK) positioning or kinematic carrier phase tracking (KCPT). After initialization, centimeter-accuracy positioning can be achieved with baselines of up to about 20 km. However, accuracy is degraded with longer baselines due to decorrelation of the ionosphere and troposphere propagation errors (see Section 8.1.1). One solution is to use multiple reference stations in analogy with RADGNSS [16].

Millimeter accuracy may be obtained over static baselines by averaging observations made over the order of an hour. High precision over long baselines can be achieved by using dual-frequency measurements, meteorological data, and long observation times to reduce the effect of differential ionosphere and troposphere errors. These techniques are normally used for surveying rather than navigation. GNSS surveying methodology is described in [17–19].

8.2.1 Integer Ambiguity Resolution

Resolution of the relative integer ambiguities, Δa_j , is the key to obtaining a carrier-phase relative positioning solution. Figure 8.2 shows the probability distributions of differential range measurements obtained from code and carrier. The carrier-based measurements are more precise but are ambiguous by an integer or half-integer number of wavelengths. Combining the two reduces the number of possible values of the integer ambiguity to the order of 10. A number of techniques may then be used, either individually or in combination, to find a unique solution. The main principles of each method are described here, while a more detailed review may be found in [20].

The simplest method is to start with a known baseline between the user and reference station. The integer ambiguities are then obtained by differencing the

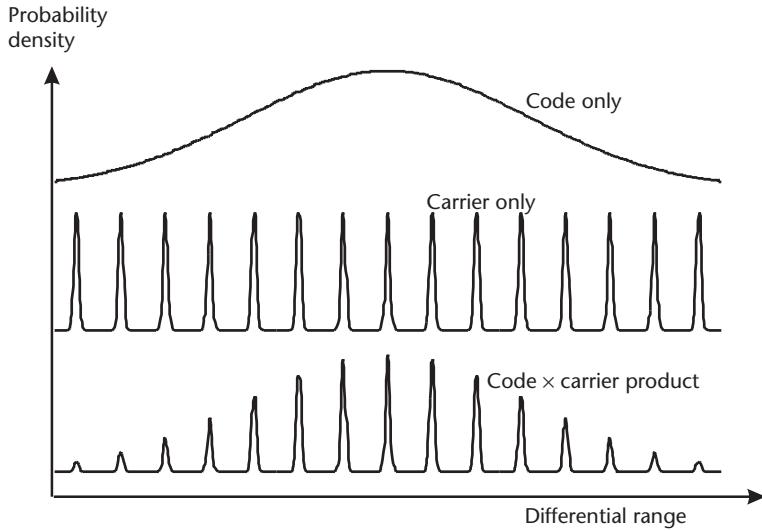


Figure 8.2 Differential range probability distributions from code and carrier measurements.

measured and predicted double differences, (8.15). This is sometimes known as receiver initialization [9].

Using a single differential code measurement to resolve the ambiguity will result in errors due to code tracking and multipath errors. However, as these errors have short correlation times and roughly zero mean, the ambiguity can be correctly resolved by using differential code measurements taken over several minutes and averaging the ambiguities obtained. For real-time applications, ADR measurements are used to account for the changes in differential pseudo-range due to user motion.

Where the receiver tracks signals from each satellite on multiple frequencies, the separation between the candidate carrier-based differential range measurements is different on each frequency, so only the candidate differential ranges where the measurements on the different frequencies align need be considered. Figure 8.3 illustrates this.

With only two frequencies, however, uncertainty in the relative ionosphere propagation delay limits the sharing of ambiguity resolution information. With three-carrier ambiguity resolution (TCAR), the ionosphere propagation delay on the third frequency can be determined from that on the others, allowing a unique ambiguity solution to be found, provided the carrier measurements are sufficiently precise [21].

A common approach with carrier measurements on two frequencies is to difference them to produce *wide lane* measurements, which have a much larger wavelength of $c/(f_{ca,a} - f_{ca,b})$, where $f_{ca,a}$ and $f_{ca,b}$ are the two carrier frequencies, reducing the number of candidate values of the integer ambiguity. However, the tracking noise is increased.

Consistency-check techniques compute sets of candidate least-squares navigation solutions with different values for the integer ambiguities. The solution with the correct values for the integer ambiguities is then the one with the smallest sum

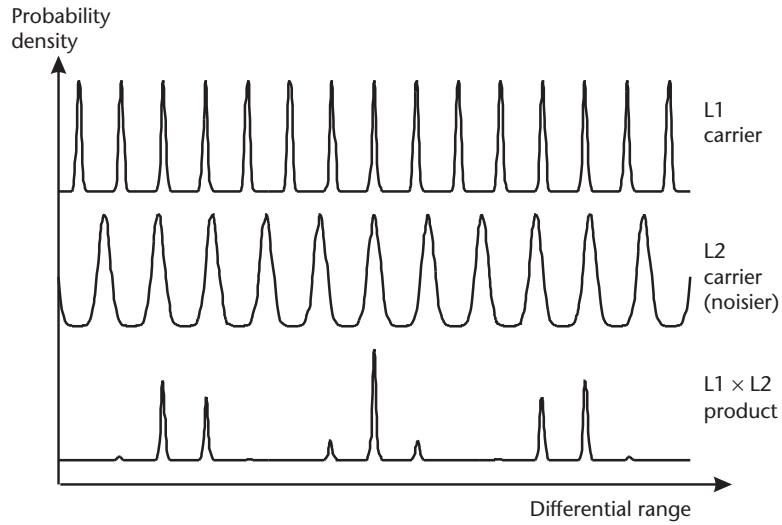


Figure 8.3 Differential range probability distributions from dual-frequency carrier measurements.

of squares of the residuals (see Section 5.5.1). At least five satellites must be tracked, and averaging across a number of measurement sets is generally required. The greater the number of satellites tracked, the quicker the ambiguities may be resolved.

Changes in the signal geometry as the satellites move can aid ambiguity resolution by consistency checks as the residuals for the false ambiguity solutions vary. It can also be used standalone, but only with a static baseline and a long observation time [20].

An efficient way of combining these ambiguity resolution techniques is to implement a float Kalman filter. This inputs both code and carrier measurements on all frequencies tracked and estimates both the differential or relative navigation solution and the ambiguities, together with a number of line-of-sight error terms. The ambiguities are estimated as real values. Once the ambiguity state uncertainties have dropped well below half a wavelength, a fixing algorithm is then used to determine a set of ambiguities consistent with the float solution. The most commonly used ambiguity-fixing algorithm is the least-squares ambiguity decorrelation adjustment (LAMBDA) method [22].

Where a sufficiently large number of satellites are tracked, the consistency-check method can provide an ambiguity fix from a single set of measurements. Otherwise, initial ambiguity resolution takes a few minutes.

Carrier-phase positioning is severely disrupted by carrier cycle slips (see Section 7.3.4) because they change the integer ambiguities. Note that half-integer cycle slips are likely with a Costas discriminator and may be followed by half-cycle corrections when the user equipment identifies sign errors in the navigation data message. Therefore, a robust carrier-phase positioning algorithm must incorporate cycle slip detection and correction. One approach is to compare the step change between successive relative carrier phase or double-difference measurements with values predicted from the range-rate measurements [23] or velocity solution.

8.2.2 GNSS Attitude Determination

Where a carrier-phase relative GNSS solution is obtained between a pair of antennas attached to the same vehicle, it can be used to obtain information about the host vehicle's attitude. As the baseline between the antennas is much smaller than the distance to the satellites, the line-of-sight vectors from a pair of antennas to a given satellite may be treated as parallel. Therefore, the angle, θ , between the baseline and the line of sight is given by $\cos \theta = \Delta \rho_{ab} / r_{ab}$, where $\Delta \rho_{ab}$ is the relative range measurement and r_{ab} is the known baseline length, as shown in Figure 8.4. The line-of-sight vector with respect to the Earth is known, so information about the host vehicle's attitude with respect to the Earth can be obtained.

More generally, if carrier-phase GNSS is used to make a measurement of the baseline in local navigation frame axes, $\tilde{\mathbf{r}}_{ab}^n$, which may be obtained from an ECEF-frame measurement using (2.31) and (2.99), this may be related to the known body-frame baseline, \mathbf{r}_{ab}^b , by

$$\tilde{\mathbf{r}}_{ab}^n = \tilde{\mathbf{C}}_b^n \mathbf{r}_{ab}^b \quad (8.16)$$

However, this does not give a unique solution to the attitude, \mathbf{C}_b^n , as the component about the baseline is undetermined; only two components may be obtained from a single baseline measurement. To resolve this, a third antenna, denoted by c , must be introduced which is noncollinear with the other two antennas, providing a second baseline. Combining the two measurements,

$$(\tilde{\mathbf{r}}_{ab}^n \quad \tilde{\mathbf{r}}_{bc}^n) = \tilde{\mathbf{C}}_b^n (\mathbf{r}_{ab}^b \quad \mathbf{r}_{bc}^b) \quad (8.17)$$

The attitude can then be obtained using [1]

$$\tilde{\mathbf{C}}_b^n = (\tilde{\mathbf{r}}_{ab}^n \quad \tilde{\mathbf{r}}_{bc}^n) (\mathbf{r}_{ab}^b \quad \mathbf{r}_{bc}^b)^T [(\mathbf{r}_{ab}^b \quad \mathbf{r}_{bc}^b) (\mathbf{r}_{ab}^b \quad \mathbf{r}_{bc}^b)^T]^{-1} \quad (8.18)$$

where further baselines may be added by extending the number of columns of the baseline matrices. Other solutions are described in [24, 25].

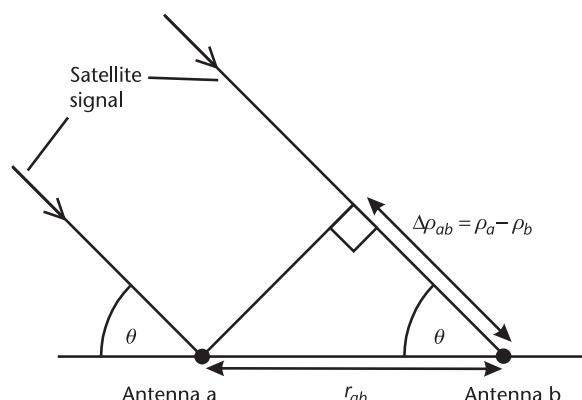


Figure 8.4 Schematic of GNSS attitude determination.

An attitude solution may be obtained with signals from only two GNSS satellites, as the known baseline lengths remove one degree of freedom from the baseline measurements, while the use of a common receiver design and shared receiver clock across all the antennas removes the relative phase offsets, provided the antenna cable lags are calibrated.

The attitude measurement accuracy is given by the ratio of the carrier phase baseline measurement accuracy to the baseline length. So, for a 1m rigid baseline, measured to a 1-cm precision, the attitude measurement standard deviation is 10 mrad (about 0.6°). Longer baselines provide greater attitude measurement precision, provided that the baseline is rigid. Flexure degrades the measurement accuracy. However, short baselines convey the advantage of fewer integer ambiguity combinations to search.

As the measurement errors are noise-like, accuracy for static applications is improved by averaging over time. For dynamic applications, noise smoothing can be achieved through integration with an INS as described in Section 12.4.3.

8.3 Poor Signal-to-Noise Environments

This section describes a number of techniques that can enable GNSS user equipment to operate in a poorer signal-to-noise environment—in other words, with a lower carrier power to noise density, C/N_0 , than standard designs. Most of these techniques may be combined. Low signal-to-noise levels can arise because of unintentional interference, deliberate jamming, or weak signals.

Sources of unintentional interference include broadcast television, mobile satellite services, ultrawideband communications, radar, cellular telephones, and DME/Tactical Air Navigation (TACAN) [26–28]. Unintentional interference will become less critical as open-access GNSS signals become available in a range of frequency bands, simply because simultaneous interference in all frequency bands is much less probable than interference in the L1 band only. However, interference from solar radio bursts can affect all GNSS signals [29]. High levels of destructive multipath interference can also be a problem.

Deliberate jamming of GNSS signals has historically been a military issue. However, with the likely advent of GNSS-based road user charging, asset tracking, and law enforcement, it is likely to become more widespread. GPS jammer designs are readily available on the Internet, while the requisite components are inexpensive.

In indoor environments, GNSS signals are typically 15–40 dB weaker than out in the open. This is due to a mixture of attenuation by the fabric of the building and rapid fading due to multipath interference between signal components of similar strength [30]. GNSS signals can also be weakened by foliage, with a typical attenuation of 1–4 dB per tree [31]. In urban canyons, there may be insufficient direct signals to compute a navigation solution, forcing users to make use of reflected signals, which are attenuated and may be LHCP, reducing the antenna gain. For space applications, signals may be received from the low-gain regions of the GNSS satellite antenna patterns.

In all weak-signal environments, the user equipment must contend with different signals having significantly different strengths, which can lead to cross-correlation problems with GPS C/A code.

8.3.1 Antenna Systems

The most effective defense against unintentional interference and deliberate jamming is a controlled-reception-pattern antenna (CRPA) system. The CRPA comprises an array of GPS antennas, mounted with their centers usually about half a wavelength apart, as illustrated by Figure 8.5. Operational CRPAs tend to comprise 4 or 7 elements, while larger arrays have been developed for experimental use. An antenna control unit (ACU) then varies the reception pattern of the antenna array by combining the signals from each antenna element with different gains and phases.

Early designs of CRPA system are null-steering, whereby the ACU acts to minimize the received RF power on the basis that unwanted interfering signals must always be stronger than the wanted GNSS signals, as the latter lie below thermal noise levels (see Section 6.1.3). This results in an antenna pattern with minima, or nulls, in the direction of each interfering source, improving the signal-to-noise level within the receiver by more than 20 dB [27, 32]. An n -element CRPA system can produce up to $n - 1$ deliberate nulls. Incidental, or parasitic, nulls also occur. A null can sometimes coincide with a GNSS satellite line of sight, attenuating signals from that satellite. Null-steering CRPA systems offer no benefit in weak-signal environments.

More advanced CRPA systems are beam-forming, whereby an antenna pattern is created with a gain maximum in the direction of the wanted satellite signal [33]. A separate antenna pattern is formed for each satellite tracked, so the receiver must have multiple front-ends. The ACU may determine the gain maxima by seeking to maximize the receiver's C/N_0 measurements. Alternatively, the CRPA attitude may be supplied by an INS and combined with satellite line-of-sight data. Beam-forming CRPA systems can potentially improve the receiver signal to noise in weak-signal environments.

CRPA systems have the drawback of being expensive, at over \$/€10,000 each, and large, with 7-element CRPAs between 14 and 30 cm in diameter. A simpler alternative for air applications is a canceller, which uses only two antennas, pointing upward and downward, and taking advantage of the fact that the GNSS signals generally come from above, while the interference generally comes from below [32].

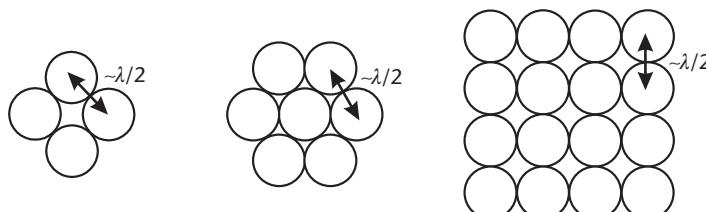


Figure 8.5 Schematic of 4-, 7-, and 16-element controlled-reception-pattern antennas.

8.3.2 Receiver Front-End Filtering

Careful design of the receiver's ADC can improve performance in weak signal-to-noise environments. An AGC prevents interference from saturating the receiver, while a larger number of quantization levels lets more signal information through [27, 28]. Pulsed interference may be mitigated using pulse blanking, whereby the ADC output is zeroed when the interference exceeds a certain margin, improving the time-averaged C/N_0 in the baseband signal processor [34].

Where the interference source has a narrower bandwidth than the wanted GNSS signal, it can be filtered by a spectral filtering technique, such as an ATF [27] or frequency-domain interference suppressor (FDIS) [35]. These use a FFT to generate a frequency-domain power spectrum and then identify which frequencies are subject to interference. The components of the signal at the interfering frequencies are then attenuated.

8.3.3 Assisted GNSS

In many poor signal-to-noise environments, GNSS signals can be acquired and tracked, but the navigation data message cannot be demodulated. Therefore, a stand-alone GNSS receiver may have to rely on out-of-date ephemeris, satellite clock, and ionosphere calibration parameters, degrading the navigation solution, while a “cold-start” navigation solution cannot be obtained at all.

One solution is assisted GNSS (AGNSS), which uses a separate communication link, such as a mobile phone system, to provide the information in the navigation data message [36]. In addition, AGNSS, also known as network assistance, can speed up acquisition by providing approximate user position (e.g., through identification of the mobile phone cell) and by calibrating the receiver clock. Accurate clock calibration is inherent in the CDMA cell phone standard, but must be added to the GSM system—for example, using the fine time aiding (FTA) technique [37]. The combination of FTA and AGNSS is known as enhanced GNSS (EGNSS). In some AGNSS systems, the navigation processor function is implemented by the network.

8.3.4 Acquisition

GNSS acquisition in poor signal-to-noise environments requires long dwell times for each cell in the search so that the signal can be identified above the noise and/or interference. For coherent integration, the required dwell time varies as $(c/n_0)^{-1}$, whereas, for noncoherent integration, it varies as $(c/n_0)^{-2}$. Thus more benefit is obtained from increasing the coherent integration time, though this does increase the number of Doppler bins to search (see Section 7.3.1) [38].

Long noncoherent integrations also require more closely spaced Doppler bins because the change in code phase over the total integration interval takes over from the change in carrier phase over the coherent integration interval as the limiting factor. The threshold is about 20 seconds for GPS C/A code and 2 seconds for P(Y) code (assuming $\tau_a = 20$ ms). The number of Doppler bins required is then proportional to the chipping rate multiplied by the dwell time.

To acquire signals within a practical time frame, one or both of two measures must be taken. The first option is to use AGNSS, including approximate user position and receiver clock calibration, to reduce the number of cells to search. User position and satellite ephemeris information without receiver clock calibration does not reduce the number of code bins to search, but does reduce the Doppler search window where the reference oscillator is a TCXO or OCXO. Note that for reacquisition of signals, the information provided by AGNSS should already be held by the receiver. The second option is to use massively parallel correlator arrays and/or FFT searches to maximize the number of cells searched in parallel (see Section 7.3.1).

A problem with acquiring weak GPS C/A-code signals is that cross-correlation peaks between the reference code and stronger C/A-code signals can be mistaken for the signal being acquired. A well-designed acquisition algorithm will acquire the strongest signals first. The on-frequency cross-correlation peaks will then be known and algorithms can be designed to eliminate them from the search. Cross-correlation peaks can also be found at 1-kHz frequency offsets and must be identified through a mismatch between carrier cycles and code chips [39].

New GNSS satellites broadcast multiple open-access signals. The effective signal to noise is improved if these signals are acquired together rather than separately, summing the Is and Qs obtained by accumulating the correlator outputs for different signals from the same satellite. Signals on different frequencies may be summed coherently, provided that combinations assuming a number of different phase offsets are tested in parallel [40].

A further issue is that, where there is insufficient C/N_0 to decode the navigation data message, it cannot be used to resolve the ambiguity in the pseudo-range measurements due to the code repetition period (see Section 7.3.7). AGNSS can be used where approximate user position and receiver clock calibration is included. Otherwise, each candidate position and clock-offset solution must be searched. With signals from five or more satellites, a consistency check can be applied, selecting the candidate least-squares navigation solution with the smallest residuals (see Section 7.5.1). With only four satellite signals, the solution must be constrained to within a certain distance of the Earth's surface. Either method easily resolves the relative ambiguity between the pseudo-ranges. However, the common-mode ambiguity is more difficult to resolve, as the candidate navigation solutions are much closer together. A false solution results in the navigation processor assuming that all of the satellites are an integer number of code repetition periods (1 ms for GPS C/A code) ahead of or behind their true positions, resulting in a meter order user position error.

Resolving this common-mode timing ambiguity requires signals from at least five satellites and averaging of residual data across several iterations. Another method is to minimize the difference between the measured and predicted pseudo-range rates as a function of the transmission time. This is more accurate where the user can be assumed to be stationary [41].

Decoding the legacy navigation data messages can be enhanced by averaging the corresponding bits in successive message cycles [38].

8.3.5 Tracking

As shown in Sections 7.3.2, 7.3.3, and 7.4.3, the code and carrier tracking-loop bandwidths are a tradeoff between noise resistance and dynamics response. Thus, tracking can be maintained in a weak signal-to-noise and low-dynamics environment by reducing the tracking-loop bandwidths. Carrier frequency tracking is more robust than carrier phase tracking. With a noncoherent code discriminator or Costas carrier discriminator, the minimum c/n_0 at which tracking can be maintained varies as the inverse square of the tracking-loop bandwidth, noting that the minimum carrier-tracking bandwidth is constrained by the reference oscillator noise. The ranging processor can be designed to adapt the tracking loop bandwidths as a function of the measured c/n_0 to maintain the optimum tradeoff between noise resistance and dynamics response, noting that this may be done implicitly where the navigation and tracking functions are combined (see Section 7.5.3). Narrow tracking-loop bandwidths may be maintained in a high-dynamics environment using aiding information from a dead reckoning system, such as an INS, as discussed in Section 12.1.4.

A limiting factor of conventional tracking is the pull-in range of the discriminator functions (see Figures 7.17, 7.21, and 7.22). The code pull-in range can be expanded by replacing the early, prompt, and late correlators by a correlator bank. However, feeding the outputs of an extended-range correlator bank into a discriminator function increases the noise, canceling the benefit of an extended pull-in range.

A solution is to feed the outputs from a bank of correlators into a limited-window acquisition algorithm with a duty cycle spanning several correlator accumulation intervals and matched to the desired tracking-loop time constant. Implementing a FFT also extends the carrier-frequency pull-in range. In this *batch processing* approach [42, 43], a tracking loop can be formed by using the output of one acquisition cycle to center the code and Doppler search window of the following cycle via the NCOs. Figure 8.6 illustrates this. The batch processing

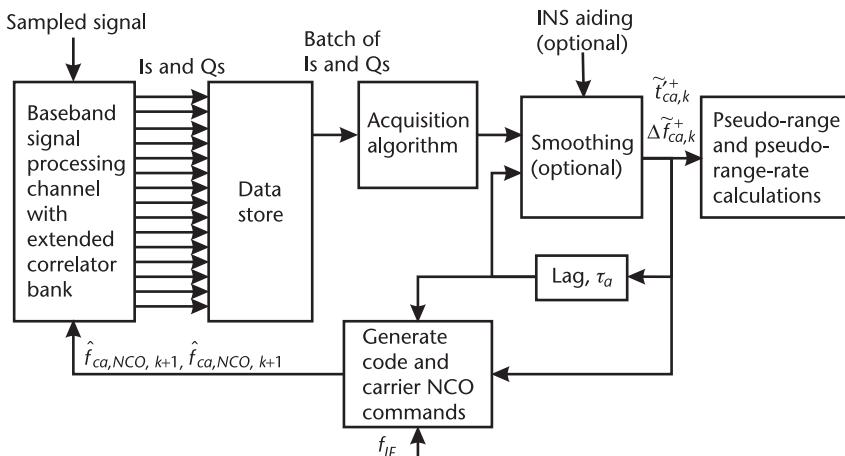


Figure 8.6 Batch-processing tracking architecture.

approach enables tracking to be maintained at lower C/N_0 levels, albeit at the cost of a higher processing load. Note that larger pseudo-range and pseudo-range-rate errors than in conventional tracking can be exhibited before tracking lock is lost.

8.3.6 Extended Coherent Integration

Acquisition and tracking performance in a poor signal-to-noise environment are optimized by maximizing the accumulation, or coherent integration, interval of the correlator I and Q outputs. For the new data-free GNSS signals, the accumulation interval is limited only by the pseudo-range-rate tracking accuracy, as discussed in Section 7.2.4.4.

For the legacy signals, modulated with a navigation data message, accumulation over more than one data-bit interval requires the receiver to multiply the reference code or correlator outputs by a locally generated copy of the message. This is known as data wipeoff. Assisted GNSS enables real-time transmission of the navigation data message as demodulated by a receiver in a strong signal-to-noise environment, while the legacy GPS and GLONASS messages are regular enough for the user equipment to predict most of their content from recently stored messages, where it is able to gather this data. Otherwise, a number of techniques have been developed for estimating the navigation data bits [38, 44–46]. Essentially, they compute candidate coherent summations with each possible data-bit combination and then use the set that gives the highest value of $(\sum I_P)^2 + (\sum Q_P)^2$.

A coherent integration time of 10 seconds has been demonstrated using a stationary antenna, assisted GPS, and a high-quality reference oscillator [47].

8.4 Multipath Mitigation

As shown in Section 7.4.4, multipath interference can produce significant errors in GNSS user-equipment code and carrier-phase measurements, with the size of the errors caused by a reflected signal of a given amplitude and lag depending on the signal type and receiver design. This section describes how antenna systems, receiver designs, mapping techniques, and filtering in the navigation processor may be used to mitigate the effects of multipath on code tracking.

8.4.1 Antenna Systems

The strength of reflected signals reaching a GNSS receiver can be minimized by careful design of the antenna system. GNSS signals are transmitted with RHCP, while reflection by a smooth surface generally reverses this to LHCP. Therefore, an antenna designed to be sensitive to RHCP signals, but not LHCP signals, can reduce multipath interference by about 10 dB. However, signals reflected by very rough surfaces are polarized randomly, so are only attenuated by 3 dB by a RHCP antenna [48].

Another characteristic of multipath environments is that most reflected signals have low or negative elevation angles. A choke-ring antenna system uses a series of concentric rings, mounted on a ground plane around the antenna element, to attenuate these signals. The integrated multipath-limiting antenna (MLA) attenuates low-elevation-angle signals further by combining a high-zenith antenna (HZA)

with a 14-element vertical antenna array [49]. The system is about 3m tall and is intended for reference stations. The effect of multipath on a reference station can also be minimized through careful siting of the antenna.

Beam-forming CRPA systems may be used to mitigate multipath by maximizing the antenna gain for direct signals [50–52]. A seven-element CRPA system reduces the pseudo-range errors due to multipath by a factor of about two, having a greater effect for high-elevation satellites.

8.4.2 Receiver-Based Techniques

A number of techniques have been developed that mitigate multipath by increasing the resolution of the code discriminator on the basis that the higher frequency components of a GNSS signal are less impacted by moderate-delay multipath interference. However, as the power in a BPSK GNSS signal is concentrated in the low-frequency components, these techniques achieve multipath mitigation at the expense of signal-to-noise performance [53].

Three techniques that replace the conventional early, prompt, and late correlators have been compared in [54]. The double-delta discriminator, also known as the Leica Type A [55], strobe correlator [56], high-resolution correlator [57], and pulse-aperture correlator, adds very early (VE) and very late (VL) correlators to conventional correlators with narrow spacing and uses them to estimate a correction to the discriminator function. The discriminator function is

$$D_{\Delta\Delta} = (I_E^2 + Q_E^2) - (I_L^2 + Q_L^2) - \frac{1}{2}(I_{VE}^2 + Q_{VE}^2) + \frac{1}{2}(I_{VL}^2 + Q_{VL}^2) \quad (8.19)$$

The early/late slope technique, or multipath elimination technology (MET) [58], places two pairs of narrowly spaced correlation channels on each side of the correlation peak and uses these to compute the slope on each side. The prompt correlation channels are then synchronized with the point where the two slopes intersect. The e1e2 technique [59] operates on the basis that multipath interference mainly distorts the late half of the correlation function (see Figure 7.27). Therefore, it places two correlation channels on the early side of the peak and acts to maintain a constant ratio between the two.

The gated-correlator method [57] retains conventional discriminators, but blanks out both the signal and the reference code away from the chip transitions to sharpen the autocorrelation function. The superresolution method [60] simply boosts the high-frequency components by filtering the spectrum of the reference code and/or the signal prior to correlation.

The multipath-estimating delay lock loop (MEDLL) samples the whole combined correlation function of the direct and reflected signals using a bank of up to 48 narrowly spaced correlators [61]. It then fits the sum of a number of idealized correlation functions to the measurements, separating the direct and reflected signal components.

The vision correlator [62] uses extra accumulators to build-up a measurement of the shape of the code-chip transitions. A number of idealized chip transitions are then fitted to the measurements, enabling the relative amplitude, lag, and phase

of the reflected components to be determined and the signal tracking corrected. This method has been shown to give better performance, particularly with short-delay multipath, than older techniques.

All of these multipath mitigation techniques require a large precorrelation bandwidth relative to the code chipping rate to obtain the best performance. There is much less scope to apply them to the high-chipping-rate GNSS signals, which are limited by the transmission bandwidth. However, the high-chipping-rate signals are only affected by short-delay multipath, which the receiver-based mitigation techniques do not compensate well. Hence, these multipath mitigation techniques effectively match the multipath performance of the low-chipping-rate signals to that of the high-chipping-rate signals at the expense of signal-to-noise performance. Thus, it is better to use high-chipping-rate signals where available.

8.4.3 Multipath Mapping

Where multipath errors are caused by reflections off the host-vehicle body, a given line-of-sight vector, resolved about the body frame, will always lead to the same multipath error. Therefore, by measuring the multipath error as a function of the line of sight, a map can be created which may then be used to calibrate out the multipath errors. This method has successfully been applied to carrier-phase multipath on a small spacecraft [63, 64]. Larger vehicles require higher resolution maps for multipath calibration, while flexure of the host-vehicle body can render a map ineffective. An alternative approach is to map which lines of sight are susceptible to multipath and down-weight these measurements in the navigation solution.

8.4.4 Navigation Processor Filtering

Multipath errors typically change every few seconds as the signal geometry changes. The longer the lag, the quicker the phase of the multipath changes. Therefore, the effect of multipath on pseudo-range measurements can be much reduced by smoothing the pseudo-range with carrier phase measurements. This may be done directly, using (7.130) or (7.142), or by implementing a filtered navigation solution, as described in Section 7.5.2, with the measurement noise covariance tuned to favor the range-rate measurements.

The navigation processor may also reject measurements contaminated by large multipath errors using integrity monitoring techniques. These are described in Chapter 15.

For ships, trains, large aircraft, and reference stations, antennas may be deployed at multiple locations and consistency checks (Section 15.4) used to determine which measurements are contaminated by multipath.

8.5 Signal Monitoring

A GNSS navigation solution can occasionally exhibit much larger errors than normal due to problems in the satellites, signal propagation environment, or user

equipment. It is essential for safety-critical applications and desirable for all applications to detect these errors and exclude all measurements from a suspect satellite or receiver channel from the navigation solution. The techniques for applying fault detection and protecting the navigation solution are known as *integrity monitoring*. User equipment-based integrity monitoring techniques are generic across different navigation systems, so are described in Chapter 15 rather than within the GNSS chapters.

Many faults are only detectable by user equipment-based integrity monitoring; however, satellite faults are more easily detected by fixed monitor stations. This is because, with the user antenna at a known location and a high-performance receiver clock, the measurement data may be focused on determining the accuracy of the incoming signals. To distinguish errors in the signals transmitted by the satellites from range measurement errors due to signal propagation, such as multipath or unusual ionosphere or troposphere behavior, a network of monitor stations in different locations should be used. Integrity monitoring stations may be combined with DGNSS reference stations. However, a separate network must be used where there is a requirement for independent monitoring of the DGNSS service. Where a fault is identified, integrity alerts may be broadcast to the user via SBAS, GBAS, the GNSS satellites themselves in the case of the Galileo safety-of-life service, or by another data link. Many applications, such as civil aviation, require users to be notified within 10 seconds of the fault occurring.

Satellite faults detectable by monitoring stations include low transmission power, clock faults, transmission of faulty navigation data, and irregular code waveforms [65]. Satellite clock faults may manifest as sudden jumps, detected through step changes in the pseudo-range measurements, or as a high drift rate, detected by observing large pseudo-range-rate errors. Faulty navigation data is detected by comparing the ephemeris, satellite clock, and ionosphere model parameters transmitted by each satellite with those computed by the monitoring network. Irregular code waveforms, known as evil waveforms, are caused by signal-generation hardware faults and distort the signal's auto-correlation function. This results in pseudo-range measurement errors that depend on the receiver design, so don't necessarily cancel in DGNSS systems. Signal quality-monitoring algorithms detect these by using receivers with multiple closely spaced correlation channels, such as the MEDLL, to observe the auto-correlation function. A vision-correlator receiver may also be used [62].

8.6 Semi-Codeless Tracking

As shown in Section 7.4.2, tracking GNSS signals on two frequencies brings significant benefits for ionosphere error calibration. At the time of writing, most GPS satellites broadcast open-access signals on one frequency only. However, advantage can be taken of the layered property of the Y code, which comprises the publicly known 10.23-Mbit s^{-1} P code multiplied by a $0.5115\text{-Mbit s}^{-1}$ encryption code. The Y code in the L2 band can be acquired and tracked by correlating it with P code, provided that the coherent integration interval of the correlator outputs is limited to the 20-bit length of each encryption code chip. Beyond this, noncoherent

accumulation must be used, summing $I^2 + Q^2$ or the root thereof. This technique is known as semi-codeless tracking [66] and brings a signal-to-noise penalty of about 18 dB over correlation with the full Y code.

References

- [1] Costentino, R. J., et al., “Differential GPS,” in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 379–452.
- [2] Parkinson, B. W., and P. K. Enge, “Differential GPS,” in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 3–50.
- [3] Kalafus, R. M., A. J. Van Dierendonck, and N. A. Pealer, “Special Committee 104 Recommendations for Differential GPS Service,” *Navigation: JION*, Vol. 33, No. 1, 1986, pp. 26–41.
- [4] SC 104, *RTCM Recommended Standards for Differential GNSS Service*, Version 3.0, RTCM, Alexandria, VA, 2004.
- [5] Kee, C., B. W. Parkinson, and P. Axelrad, “Wide Area Differential GPS,” *Navigation: JION*, Vol. 38, No. 2, 1991, pp. 123–145.
- [6] Ashkenazi, V., et al., “Wide-Area Differential GPS: A Performance Study,” *Navigation: JION*, Vol. 40, No. 3, 1993, pp. 297–319.
- [7] Kee, C., “Wide Area Differential GPS,” in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 81–115.
- [8] Muellerschoen, R. J., et al., “Real-Time Precise-Positioning Performance Evaluation of Single-Frequency Receivers Using NASA’s Global Differential GPS System,” *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1872–1880.
- [9] El-Rabbany, A., *Introduction to GPS: The Global Positioning System*, 2nd ed., Norwood, MA: Artech House, 2006.
- [10] Sharpe, T., R. Hatch, and F. Nelson, “John Deere’s StarFire System: WADGPS for Precision Agriculture,” *Proc. ION GPS 2000*, Salt Lake City, UT, September 2000, pp. 2269–2277.
- [11] Gao, Y., “Precise Point Positioning and Its Challenges,” *Inside GNSS*, November–December 2006, pp. 16–18.
- [12] Moore, A. W., “The International GNSS Service—Any Questions?” *GPS World*, January 2007, pp. 58–64.
- [13] Pocknee, S., et al., “Experiences with the OmiSTAR HP Differential Correction Service on an Autonomous Agricultural Vehicle,” *Proc. ION 60th AM*, Dayton, OH, June 2004, pp. 346–353.
- [14] Dixon, K., “StarFireTM: A Global SBAS for Sub-Decimeter Precise Point Positioning,” *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 2286–2296.
- [15] Ward, P. W., J. W. Betz, and C. J. Hegarty, “Satellite Signal Acquisition, Tracking and Data Demodulation,” in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 153–241.
- [16] Raquet, J., G. Lachapelle, and T. Melgård, “Test of a 400 km × 600 km Network of Reference Receivers for Precise Kinematic Carrier-Phase Positioning in Norway,” *Proc. ION GPS-98*, Nashville, TN, September 1998, pp. 407–416.
- [17] Leick, A., *GPS Satellite Surveying*, 3rd ed., New York: Wiley, 2004.
- [18] Hoffmann-Wellenhof, B., H. Lichtenegger, and J. Collins, *Global Positioning Systems*, 5th ed., Vienna, Austria: Springer, 2001.

- [19] Goad, C., "Surveying with the Global Positioning System," in *Global Positioning System: Theory and Applications Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 501–517.
- [20] Misra, P., and P. Enge, *Global Positioning System Signals, Measurements, and Performance*, Lincoln, MA: Ganga-Jamuna Press, 2001.
- [21] Hatch, R. R., "A New Three-Frequency, Geometry-Free, Technique for Ambiguity Resolution," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 309–316.
- [22] Teunissen, P. J. G., P. J. De Jonge, and C. C. J. M. Tiberius, "Performance of the LAMBDA Method for Fast GPS Ambiguity Resolution," *Navigation: JION*, Vol. 44, No. 3, 1997, pp. 373–383.
- [23] Kim, D., and R. B. Langley, "Instantaneous Real-Time Cycle-Slip Correction for Quality Control of GPS Carrier-Phase Measurements," *Navigation: JION*, Vol. 49, No. 4, 2002, pp. 205–222.
- [24] Cohen, C. E., "Attitude Determination," in *Global Positioning System: Theory and Applications Volume II*, B. W. Parkinson and J. J. Spilker, Jr. (eds.), Washington, D.C.: AIAA, 1996, pp. 518–538.
- [25] Farrell, J. A., and M. Barth, *The Global Positioning System and Inertial Navigation*, New York: McGraw-Hill, 1999.
- [26] Carroll, J., et al., *Vulnerability Assessment of the Transportation Infrastructure Relying on the Global Positioning System*, John A. Volpe National Transportation Systems Center report for U.S. Department of Transportation, 2001.
- [27] Spilker, J. J., Jr., and F. D. Natali, "Interference Effects and Mitigation Techniques," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 717–771.
- [28] Ward, P. W., J. W. Betz, and C. J. Hegarty, "Interference, Multipath and Scintillation," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 243–299.
- [29] Cerruti, A., "Observed GPS and WAAS Signal-to-Noise Degradation Due to Solar Radio Bursts," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 1369–1376.
- [30] Haddrell, T., and A. R. Pratt, "Understanding The Indoor GPS Signal," *Proc. ION GPS 2001*, Salt Lake City, UT, September 2001, pp. 1487–1499.
- [31] Spilker, J. J., Jr., "Foliage Attenuation for Land Mobile Users," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 569–583.
- [32] Rounds, S., "Jamming Protection of GPS Receivers, Part II: Antenna Enhancements," *GPS World*, February 2004, pp. 38–45.
- [33] Owen, J. I. R., and M. Wells, "An Advanced Digital Antenna Control Unit for GPS," *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 402–407.
- [34] Hegarty, C., et al., "Suppression of Pulsed Interference Through Blanking," *Proc. ION 56th AM*, San Diego, CA, June 2000, pp. 399–408.
- [35] Capozza, P. T., et al., "A Single-Chip Narrow-Band Frequency-Domain Excisor for a Global Positioning System (GPS) Receiver," *IEEE Journal of Solid-State Circuits*, Vol. 35, No. 3, 2000, pp. 401–411.
- [36] Bullock, J. B., et al., "Integration of GPS with Other Sensors and Network Assistance," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 459–558.
- [37] Duffett-Smith, P., and T. Pratt, "Fine Time Aiding in Unsynchronised Cellular Systems: The Benefits for GPS Integration in Mobile Phones," *Proc. ION NTM*, Monterey, CA, January 2006, pp. 159–166.
- [38] Ziedan, N. I., *GNSS Receivers for Weak Signals*, Norwood, MA: Artech House, 2006.
- [39] Mattos, P. G., "High Sensitivity GNSS Techniques to Allow Indoor Navigation with GPS and with Galileo," *Proc. GNSS 2003*, ENC, Graz, Austria, April 2003.

- [40] Ioannides, R. T., L. E. Aguado, and G. Brodin, "Coherent Integration of Future GNSS Signals," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 1253–1268.
- [41] Sheynblat, L., and J. C. Scheller, Jr., "Method and Apparatus for Determining Time in a Satellite Positioning System," U.S. patent 6,215,442, filed 1997, granted 2001.
- [42] Van Graas, F., et al., "Comparison of Two Approaches for GNSS Receiver Algorithms: Batch Processing and Sequential Processing Considerations," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 200–211.
- [43] Anyaegbu, E., "A Frequency Domain Quasi-Open Tracking Loop for GNSS Receivers," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 790–798.
- [44] Soloviev, A., F. Van Graas, and S. Gunawardena, "Implementation of Deeply Integrated GPS/ Low-Cost IMU for Reacquisition and Tracking of Low CNR GPS Signals," *Proc. ION NTM*, San Diego, CA, January 2004, pp. 923–935.
- [45] Ziedan, N. I., and J. L. Garrison, "Unaided Acquisition of Weak GPS Signals Using Circular Correlation or Double-Block Zero Padding," *Proc. IEEE PLANS*, Monterey, CA, April 2004, pp. 461–470.
- [46] Psiaki, M. L., and H. Jung, "Extended Kalman Filter Methods for Tracking Weak GPS Signals," *Proc. ION GPS 2002*, Portland, OR, September 2002, pp. 2539–2553.
- [47] Watson, W., et al., "Investigating GPS Signals Indoors with Extreme High-Sensitivity Detection Techniques," *Navigation: ION*, Vol. 52, No. 4, 2005, pp. 199–213.
- [48] Braasch, M. S., "Multipath Effects," in *Global Positioning System: Theory and Applications, Volume I*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 547–568.
- [49] Thornberg, D. B., et al., "LAAS Integrated Multipath-Limiting Antenna," *Navigation: ION*, Vol. 50, No. 2, 2003, pp. 117–130.
- [50] Brown, A., and N. Gerein, "Test Results from a Digital P(Y) Code Beamsteering Receiver for Multipath Minimization," *Proc. ION 57th AM*, Albuquerque, NM, June 2001, pp. 872–878.
- [51] Weiss, J. P., et al., "Analysis of P(Y) Code Multipath for JPALS LDGPS Ground Station and Airborne Receivers," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 2728–2741.
- [52] McGraw, G. A., et al., "GPS Multipath Mitigation Assessment of Digital Beam Forming Antenna Technology in a JPALS Dual Frequency Smoothing Architecture," *Proc. ION NTM*, San Diego, CA, January 2004, pp. 561–572.
- [53] Pratt, A. R., "Performance of Multi-Path Mitigation Techniques at Low Signal to Noise Ratios," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 43–53.
- [54] Irsigler, M., and B. Eissfeller, "Comparison of Multipath Mitigation Techniques with Consideration of Future Signal Structures," *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 2584–2592.
- [55] Hatch, R. R., R. G. Keegan, and T. A. Stansell, "Leica's Code and Phase Multipath Mitigation Techniques," *Proc. ION NTM*, January 1997, pp. 217–225.
- [56] Garin, L., and J.-M. Rousseau, "Enhanced Strobe Correlator Multipath Rejection for Code and Carrier," *Proc. ION GPS-97*, Kansas, MO, September 1997, pp. 559–568.
- [57] McGraw, G. A., and M. S. Braasch, "GNSS Multipath Mitigation Using Gated and High Resolution Correlator Concepts," *Proc. ION GPS-99*, Nashville, TN, September 1999, pp. 333–342.
- [58] Townsend, B., and P. Fenton, "A Practical Approach to the Reduction of Pseudorange Multipath Errors in a L1 GPS Receiver," *Proc. ION GPS-94*, Salt Lake City, UT, September 1994, pp. 143–148.
- [59] Mattos, P. G., "Multipath Elimination for the Low-Cost Consumer GPS," *Proc. ION GPS-96*, Kansas City, MO, September 1996, pp. 665–672.
- [60] Weill, L. R., "Application of Superresolution Concepts to the GPS Multipath Mitigation Problem," *Proc. ION NTM*, Long Beach, CA, January 1998, pp. 673–682.

- [61] Townsend, B. R., "Performance Evaluation of the Multipath Estimating Delay Lock Loop," *Proc. ION NTM*, Anaheim, CA, January 1995.
- [62] Fenton, P. C., and J. Jones, "The Theory and Performance of NovAtel Inc's Vision Correlator," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 2178–2186.
- [63] Reichert, A. K., and P. Axelrad, "Carrier-Phase Multipath Corrections for GPS-Based Satellite Attitude Determination," *Navigation: JION*, Vol. 48, No. 2, 2001, pp. 77–88.
- [64] Hodgart, S., and R. Wong, "Statistically Optimized In-Flight Estimation of GPS Carrier Phase Multipath for LEO Satellite Attitude Determination," *Navigation: JION*, Vol. 53, No. 3, 2006, pp. 181–202.
- [65] Xie, G., et al., "Integrity Design and Updated Test Results for the Stanford LAAS Integrity Monitor Testbed," *Proc. ION 57th AM*, Albuquerque, NM, June 2001, pp. 681–693.
- [66] Woo, K. T., "Optimum Semicodeless Carrier-Phase Tracking on L2," *Navigation: JION*, Vol. 47, No. 2, 2000, pp. 82–99.

Terrestrial Radio Navigation

This chapter describes the terrestrial radio navigation systems in use or under development at the time of this writing. Terrestrial radio navigation history is summarized in Section 1.3.1.

Section 9.1 describes the point-source systems: nondirectional beacons (NDBs), VOR, and DME, while Section 9.2 describes Loran. These long-range navigation systems predate GNSS and now act as a backup, as they don't share the common failure modes of the different GNSS systems. They only provide horizontal position fixes, as the signal geometry (see Section 7.1.4) in the vertical axis is generally poor because the transmitters and receiver are usually close to coplanar.

Section 9.3 describes the instrument landing system, which is dedicated to aircraft landing. Section 9.4 reviews a number of relatively new techniques, designed to operate primarily in urban areas and indoors, where GNSS performance is poor. These include use of mobile phones, signals of opportunity, and UWB positioning. Sections 9.5 and 9.6 discuss relative navigation and tracking, respectively, while Section 9.7 discusses sonar positioning for use underwater.

9.1 Point-Source Systems

Point-source navigation systems can provide horizontal positioning using measurements from only one station, though multiple stations may be used. NDBs, VOR, and DME are described [1, 2].

Nondirectional beacons broadcast omnidirectional signals between 150 and 1,700 kHz. Some NDBs also broadcast radio stations or transmit LADGNSS information (see Section 8.1.2). Using a direction-finding receiver, a bearing to the beacon, accurate to about 5° , may be measured. A rough position fix may be obtained from two NDBs. Alternatively, an aircraft may use the bearing measurements to fly toward the beacon and then use the vertical null in the beam's transmission pattern to detect when it is overhead, obtaining a position fix. Dedicated NDBs are scheduled to be phased out in the near future.

VOR and DME beacons are usually co-located. They are designed to serve aircraft, so the coverage radius is typically around 400 km at high altitude but drops down to about 75 km at 300m above ground level.

VOR beacons transmit in the 108–118-MHz band. Each station transmits a 30-Hz amplitude-modulation (AM) signal, a 30-Hz frequency-modulation (FM) signal on a subcarrier, an identification code, and an optional voice signal. The relative phase of the AM and FM signals varies with azimuth. By measuring this,

VOR receivers can obtain their heading, with respect to magnetic north, from the beacon to an accuracy of 1–2°, which corresponds to a 7–14-km position accuracy at maximum range.

DME is a two-way ranging system, operating in the 960–1,215-MHz band. The user equipment, known as an interrogator, transmits a double pulse. The DME beacon, known as a transponder, then broadcasts a response double pulse on a separate frequency 50 µs after receiving the interrogator's signal. Each transponder is designed to serve 100 users at a time. Where signals from two interrogators are received in close succession, the transponder can only respond to the earlier signal. Random intervals between successive interrogations prevent repeated clashes between any pair of users.

Each DME transponder transmits pulses in response to many users. The interrogator must identify which are in response to its own signals. Initially, the range to the transponder is unknown, so the interrogator operates in search mode, where it may emit up to 150 pulse pairs per second. It listens for a response at a fixed interval from transmission, changing this interval every few pulses. When the interval corresponds to the response time, pulses will be received in response to most interrogations. Otherwise, pulses will only be received occasionally, as the responses to the other users are uncorrelated with the interrogator's transmissions. Figure 9.1 illustrates this. Once the response time has been found, the interrogator switches to track mode, dropping its interrogation rate to within 30 pulse pairs per second and only scanning either side of the predicted response time. Historically, the range error standard deviation is 130m, but with modern beacons and receivers, an accuracy of order 20m should be attainable [3].

The receiver-transmitter height difference must be accounted for in determining the horizontal range from the measured range, as Figure 9.2 illustrates. The user latitude and longitude are approximately

$$\begin{aligned}\tilde{L}_a &\approx L_t + \frac{\sqrt{\tilde{p}^2 - (\hat{h}_a - h_t)^2} \cos(\tilde{\psi}_{mp} + \hat{\alpha}_{nm})}{[R_N(L_t) + h_t]} \\ \tilde{\lambda}_a &\approx \lambda_t + \frac{\sqrt{\tilde{p}^2 - (\hat{h}_a - h_t)^2} \sin(\tilde{\psi}_{mp} + \hat{\alpha}_{nm})}{\{[R_E(L_t) + h_t] \cos L_t\}}\end{aligned}\quad (9.1)$$

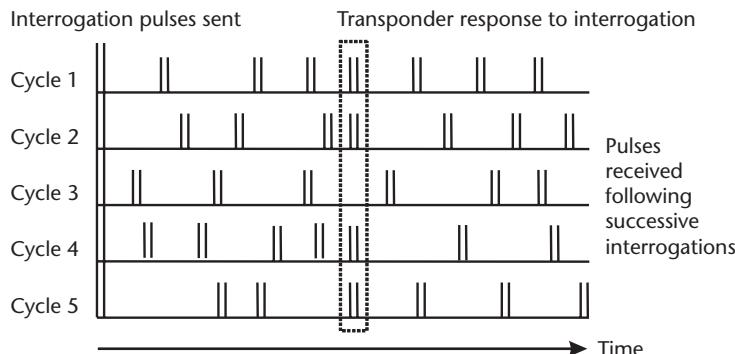


Figure 9.1 DME pulses received following successive interrogations. (After: [1].)

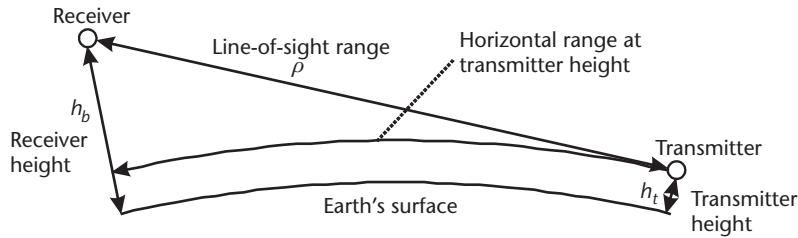


Figure 9.2 Relationship between line-of-sight and horizontal range.

where \tilde{p} is the DME range measurement; $\tilde{\psi}_{mp}$ is the VOR bearing; $\hat{\alpha}_{nm}$ is the magnetic declination (see Section 10.1.2); L_t , λ_t , and h_t are the beacon latitude, longitude, and height; and h_a is the user height. This neglects Earth curvature and the Sagnac effect (see Section 7.1.2), introducing errors of up to 65m and 0.6m, respectively. Many VOR/DME users travel along airways linking the beacons and navigate in terms of distance and bearing from their destination beacon rather than Earth-referenced position.

In most parts of Europe and North America, multiple VOR/DME beacons are within range, while greater positioning accuracy is generally available from two DME measurements than from a DME and VOR measurement. Therefore, newer DME interrogators can track multiple transponders, while VOR is scheduled to be discontinued in the near future.

TACAN is a U.S. military enhancement to DME that adds 135 pulses per second, carrying bearing information with an accuracy of 0.5° to the standard transponder transmissions. VORTAC beacons serve both VOR/DME and TACAN users.

9.2 Loran

Long-range navigation (Loran) comprises a family of systems, many of which are now obsolete. The current systems, Loran-C, Chayka, and ELoran (commonly written as eLoran), operate at 100 kHz in the low frequency (long wave) region of the spectrum. The transmitters form chains, each comprising a master and 2–5 secondary stations. Some transmitters, known as dual rates, belong to two chains [1, 2].

Traditionally, receivers have measured time differences (TDs), each comprising the difference between the times of arrival (TOAs) of signals from two transmitters in the same chain. Each TD, corrected for the difference in time of transmission (TOT), defines a hyperbola-shaped line of position (LOP) along which the receiver may be located, as Figure 9.3 illustrates. Hence, this method is known as hyperbolic positioning. A two-dimensional position fix can usually be obtained from two TDs.

Loran employs ground-wave propagation, which, at low frequencies, enables long range to be achieved independently of altitude. With modern user equipment, transmitter ranges are 1,000–1,700 km over land and 1,700–2,400 km over sea [2, 4]. Sky-wave signals travel further, but are unreliable and inaccurate. A major

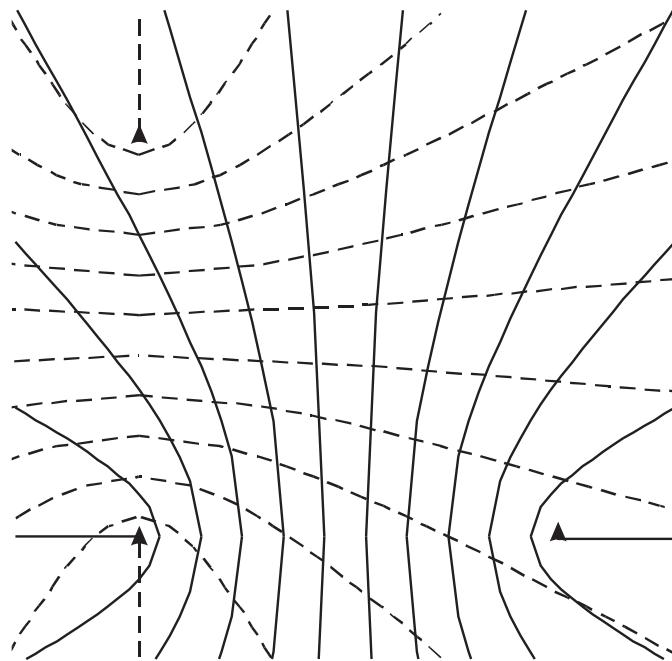


Figure 9.3 Hyperbolic lines of position from two pairs of transmitters.

advantage of Loran signals, compared to GNSS, is that they penetrate well into valleys, urban canyons, and buildings, even basements [5].

The accuracy specification for Loran-C is a 95 percent probability of a radial error within 400m over water. However, differential ELoran can achieve accuracies of order 10m at selected locations, such as airports and harbors.

This section begins with an overview of the three Loran systems, followed by descriptions of the signals and user equipment processing, positioning methods, and error sources. It concludes with a summary of differential Loran.

9.2.1 The Loran Systems

Loran-C was developed in the late 1950s as a U.S. military system, covering the North Atlantic, North Pacific, and Mediterranean. From 1974, Loran-C was made available for civil marine use and was later adopted by the U.S. aviation community. At the end of 1994, U.S. Loran stations overseas were transferred to their host countries, most of whom have maintained and developed the network.

Chayka was developed by the Soviet Union, also as a military system, and is almost identical to Loran-C.

In 2007, the combined coverage of Loran-C and Chayka comprised much of the Northern Hemisphere, including the conterminous United States, parts of Canada and Alaska, Northern Europe, the Middle East including Saudi Arabia, parts of Russia, Japan, Korea, the Chinese coast, and parts of India.

ELoran is a program of backward-compatible improvements to the Loran-C system, commencing in the mid-1990s and ongoing at the time of this writing [5, 6]. The first stage, comprising the replacement of obsolescent transmitters and

the synchronization of their TOTs to UTC, has been completed in the United States, northwest Europe, and the Far East. TOT synchronization enables cross-chain TDs and single-transmitter pseudo-ranges to be measured. By removing the requirement to track at least two transmitters in a given chain, effective coverage is improved.

The second stage of station upgrades is the transmission of differential corrections and other data using the Loran data channel (LDC) or Eurofix. Addition of the LDC to U.S. Loran stations began in 2005.

User equipment improvements in recent years include tracking of all available signals, which can be more than 30, increased sensitivity, improved databases of signal propagation delays, and magnetic (H) field antennas, which are smaller and eliminate precipitation static interference [4, 7].

9.2.2 Signals and User-Equipment Processing

Loran signals are all transmitted on a 100-kHz carrier with a 20-kHz double-sided bandwidth and vertical polarization. Stations within a chain transmit in turn, a form of TDMA. Figure 9.4 shows the signals received from one chain [2]. Each transmission comprises a group of eight 500- μ s pulses, starting 1 ms apart, with master stations adding an additional pulse 2 ms after the eighth pulse. Some stations also broadcast a ninth pulse, 1 ms after the eighth, to carry the LDC. The polarity of each pulse within a group is varied to produce a phase code, which repeats every two groups. Secondary stations use a different phase code to the master.

Each transmitter repeats at a constant interval between 50 and 100 ms, known as the group repetition interval (GRI). The GRI is different for each chain so is used as identification. Signals from different Loran chains can potentially interfere. Careful selection of the GRIs keeps the repetition intervals of the cross-chain interference patterns in excess of 10 seconds. Furthermore, modern Loran user equipment can predict which pulses will be subject to interference from other Loran stations, so can ignore them or even subtract a replica of the unwanted signal. Signals more than 40 dB weaker than the strongest available signal can thus be tracked [4].

The received signal pulses can be distorted by sky waves from the same transmitter, a form of multipath. Older receivers use a tracking point about 30 μ s from the start of the pulse. This is less than the delay of the sky wave with respect to the ground wave, ensuring that only the latter is used. Modern receivers take

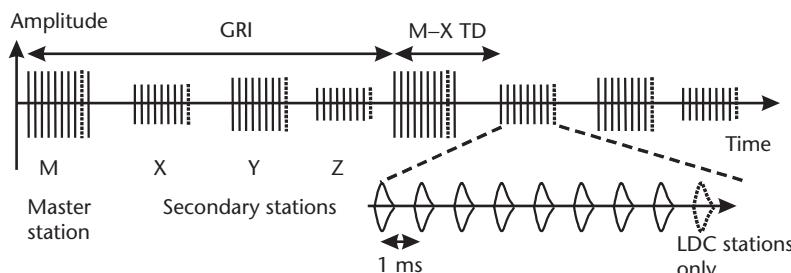


Figure 9.4 Loran signals received from one chain. (After: [2].)

multiple samples of each pulse and then process them to separate the ground-wave component.

Loran-C transmitters indicate that their signals are faulty and should not be used by a process known as blinking, whereby the first two pulses are omitted for 3.75 of every 4 seconds. Faulty ELooran transmitters are simply switched off.

Each LDC pulse is modulated with one of four delay states and eight phase states. These combine to give 5 bits of data per pulse, providing a data rate of 50–100 bit s⁻¹, depending on the GRI [8].

The Eurofix system [9], currently operating in northwest Europe and Saudi Arabia, offsets the timing of each pulse by 0, +1, or -1 μs to provide a data channel. Eurofix carries GNSS differential corrections and other data, which may include differential Loran. The average timing of each pulse group remains the same to minimize ranging distortion. The data rate is 7 bits per GRI, giving 70–140 bit s⁻¹, some of which is used for error correction.

Modern Loran user equipment operates in a similar manner to GNSS user equipment, described in Chapter 7. The signal samples are correlated with receiver-generated replicas of the expected signals, then input to signal acquisition and tracking functions, which produce pseudo-range measurements. Like GNSS, a higher signal to noise is needed for acquisition than for tracking.

9.2.3 Positioning

ELoran corrected pseudo-range measurements, $\tilde{\rho}_C$, are obtained from the measured TOA, \tilde{t}_{sa} , using

$$\tilde{\rho}_{Cj} = (\tilde{t}_{sa,j} - t_{st,j})c + \Delta\rho_{ASF,j} \quad (9.2)$$

where t_{st} is the time of transmission, c is the effective propagation speed over saltwater, $\Delta\rho_{ASF}$ is a (negative) propagation delay correction, to account for lower propagation speeds over land and other delays, and the index j denotes the Loran station. The propagation delay, known as an additional secondary factor (ASF), is corrected using a database. It is analogous to the ionosphere and troposphere corrections applied to GNSS pseudo-ranges but exhibits much less time variation.

For land and marine users, the signal measured by the user equipment is the ground wave, which follows a great circle path. Consequently, the pseudo-range may be expressed in terms of the transmitter latitude, L_t , and longitude, λ_t , and the user latitude, \hat{L}_a , and longitude, $\hat{\lambda}_a$, solution by

$$\begin{aligned} \tilde{\rho}_{Cj} = & \sqrt{[L_{tj}R_N(L_{tj}) - \hat{L}_aR_N(\hat{L}_a)]^2 + [\lambda_{tj}R_E(L_{tj}) \cos L'_{tj} - \hat{\lambda}_aR_E(\hat{L}_a) \cos L_{ta}]^2} \\ & + \delta\hat{\rho}_{rc} + \delta\rho_{ej}^+ \end{aligned} \quad (9.3)$$

where $\delta\hat{\rho}_{rc}$ is the receiver clock offset, $\delta\rho_{ej}^+$ is the measurement residual, R_N and R_E are, respectively, the north-south and east-west great circle radii of curvature, given by (2.65) and (2.66), and

$$\cos L' = \sqrt{1 - \frac{R_N^2(L)}{R_E^2(L)}(1 - \cos^2 L)} \quad (9.4)$$

is the angle subtended at the center of the Earth by an east-west great circle that subtends a unit longitude change. Note that the Sagnac effect is assumed to be accounted for in $\Delta\rho_{ASF}$.

At least three pseudo-range measurements are needed to solve for the latitude, longitude, and clock offset. With more measurements, the solution is overdetermined. A solution may be obtained by least-squares or using a Kalman filter in analogy with the GNSS navigation solution, described in Section 7.5.

A least-squares solution from n measurements is

$$\begin{pmatrix} \hat{L}_a \\ \hat{\lambda}_a \\ \hat{\delta\rho}_{rc} \end{pmatrix} = \begin{pmatrix} L_a^P \\ \lambda_a^P \\ \delta\rho_{rc}^P \end{pmatrix} + \begin{pmatrix} 1/R_N(L_a^P) & 0 & 0 \\ 0 & 1/[R_E(L_a^P) \cos L'_a] & 0 \\ 0 & 0 & 1 \end{pmatrix} (\mathbf{G}_n^T \mathbf{G}_n)^{-1} \mathbf{G}_n^T \begin{pmatrix} \tilde{\rho}_{C1} - \rho_{C1}^P \\ \tilde{\rho}_{C2} - \rho_{C2}^P \\ \vdots \\ \tilde{\rho}_{Cn} - \rho_{Cn}^P \end{pmatrix} \quad (9.5)$$

where the superscript P denotes the predicted value, and the geometry matrix is

$$\mathbf{G}_n = \begin{pmatrix} -\mathbf{u}_{at,1,N}^n & -\mathbf{u}_{at,1,E}^n & 1 \\ -\mathbf{u}_{at,2,N}^n & -\mathbf{u}_{at,2,E}^n & 1 \\ \vdots & \vdots & \vdots \\ -\mathbf{u}_{at,n,N}^n & -\mathbf{u}_{at,n,E}^n & 1 \end{pmatrix} \quad (9.6)$$

where \mathbf{u}_{at}^n is the line-of-sight unit vector from the user antenna to the transmitter, resolved in the local navigation frame.

For airborne users, the navigation solution is more complex as the signal propagation is a mixture of ground wave and line of sight, so (9.3) is not valid.

Legacy Loran user equipment measures the TD, $\Delta\tilde{E}_{TD,ij} = \tilde{t}_{sa,j} - \tilde{t}_{sa,i}$, from which a corrected delta-range, $\Delta\tilde{\rho}_C$, may be obtained using

$$\Delta\tilde{\rho}_{Cij} = (\Delta\tilde{t}_{TD,ij} - \Delta t_{NED,ij})c + \Delta\rho_{ASF,j} - \Delta\rho_{ASF,i} \quad (9.7)$$

where $\Delta t_{NED,ij} = t_{st,ij} - t_{st,l}$ is the transmission time difference, known as the nominal emission delay (NED), and the indexes i and j denote the master and secondary Loran stations, respectively.

Hyperbolic positioning does not require a receiver clock solution, so for TDs between a master, m , and n secondary stations, the least-squares solution for land and marine users is

$$\begin{pmatrix} \hat{L}_a \\ \hat{\lambda}_a \end{pmatrix} = \begin{pmatrix} L_a^P \\ \lambda_a^P \end{pmatrix} \quad (9.8)$$

$$+ \begin{pmatrix} 1/R_N(L_a^P) & 0 \\ 0 & 1/[R_E(L_a^P) \cos L_a^P] \end{pmatrix} (\mathbf{G}_n^T \mathbf{G}_n)^{-1} \mathbf{G}_n^T \begin{pmatrix} \Delta \tilde{\rho}_{Cm1} - \rho_{C1}^P + \rho_{Cm}^P \\ \Delta \tilde{\rho}_{Cm2} - \rho_{C2}^P + \rho_{Cm}^P \\ \vdots \\ \Delta \tilde{\rho}_{Cmn} - \rho_{Cn}^P + \rho_{Cm}^P \end{pmatrix}$$

where

$$\mathbf{G}_n = \begin{pmatrix} u_{at,m,N}^n - u_{at,1,N}^n & u_{at,m,E}^n - u_{at,1,E}^n \\ u_{at,m,N}^n - u_{at,2,N}^n & u_{at,m,E}^n - u_{at,2,E}^n \\ \vdots & \vdots \\ u_{at,m,N}^n - u_{at,n,N}^n & u_{at,m,E}^n - u_{at,n,E}^n \end{pmatrix} \quad (9.9)$$

9.2.4 Error Sources

The main cause of biases in Loran pseudo-range and TD measurements is variation in the signal propagation speed, primarily over land, where it is about 0.5 percent lower than over water. Thus, for a 1,000-km land path, $\Delta \rho_{ASF}$ is around -5 km. The propagation speed varies with terrain conductivity and roughness, making it difficult to model. The ASF corrections supplied with the original Loran-C system were only accurate to a few hundred meters [2]. For ELoran, higher resolution ASF databases are being developed, using a mixture of measurements and modeling, which reduce the range biases to within 100m [10]. Higher precision ASF corrections may be obtained for areas of interest, such as airports and harbors, by using a higher resolution grid than applied generally [11].

The ASFs vary over time due to the weather. Variations over the course of a day are of order 3m (1σ) where the TOTs are synchronized to UTC [4]. However, summer-to-winter variations can be several tens of meters where land freezes and thaws. Seasonal changes can be incorporated in the ASF corrections, but the accuracy is limited by variations in the timing and severity of the seasons each year. To fully calibrate the time-varying biases, either differential Loran (Section 9.2.5) or Loran integrated with other positioning systems, such as GNSS, as discussed in Section 14.2, must be used.

Like any radio system, Loran signals are subject to multipath. Due to the long wavelength, this arises only from large objects, such as mountains, bridges, and transmission lines. Where high-resolution databases are used, the effect of multipath can be incorporated into the ASF corrections [5] as it is correlated over several hundred meters and exhibits little time variation as the transmitters are fixed. In contrast to GNSS, multipath does not lead to poorer performance in urban areas.

Random errors in Loran measurements are due to RF noise arising from atmospheric electrical noise, man-made interference, and the other Loran signals. With

a strong signal and modern user equipment, the range error standard deviation due to noise can be as low as 1.5m [7], while the noise with a weak signal can be around 100m. Thus, with all-in-view user equipment, it is important to weight each measurement appropriately in the navigation solution. As with GNSS (see Section 7.4.3), there is a design tradeoff between noise performance and dynamics response. Loran user-equipment tracking loops typically have time constants of several seconds, so position errors of tens of meters can potentially arise from delays in responding to changes in host vehicle velocity [12]. Therefore, the user equipment design should be carefully matched to the application.

As with GNSS, the dilution of precision can be used to predict the position accuracy from the pseudo-range or TD accuracy. The discussion in Section 7.1.4 is largely applicable to Loran, except that there is no vertical component. With pseudo-range measurements, the DOPs are given by

$$\begin{pmatrix} D_N^2 & \cdot & \cdot \\ \cdot & D_E^2 & \cdot \\ \cdot & \cdot & D_T^2 \end{pmatrix} = (\mathbf{G}_n^T \mathbf{G}_n)^{-1} \quad (9.10)$$

where \mathbf{G}_n is given by (9.6), while for hyperbolic positioning,

$$\begin{pmatrix} D_N^2 & \cdot \\ \cdot & D_E^2 \end{pmatrix} = (\mathbf{G}_n^T \mathbf{G}_n)^{-1} \quad (9.11)$$

where \mathbf{G}_n is given by (9.9). Note that DOPs calculated for all-in-view user equipment can be overoptimistic due to correlations in residual propagation delays between signals.

9.2.5 Differential Loran

Like differential GNSS (Section 8.1), differential Loran is designed to eliminate time-varying signal propagation delays and TOT errors by measuring these at a reference station at a known location and then transmitting corrections to the users. The corrections comprise the difference between the smoothed pseudo-range or TD measurements and their nominal values. An ASF corrections database is used to account for the spatial variation in propagation delays as normal. The reference station and all users must use the same database.

Differential corrections may be transmitted using the ELooran LDC, Eurofix or a private link. Because of variation in the ground-wave propagation characteristics over different terrain, the spatial decorrelation is higher than for DGNSS, so best results are obtained within about 30 km of the reference station [6]. The position error standard deviation is about 10m [5].

9.3 Instrument Landing System

The ILS is used for guiding aircraft approaches to runways. Signals are transmitted over a cone, centered about the recommended approach route and extending for

10–20 km from the runway threshold. There are three components, the localizer, the glide slope, and up to three marker beacons, as Figure 9.5 illustrates [2, 13].

The localizer, transmitting in the 108–112-MHz band, extends to at least $\pm 35^\circ$ either side of the runway centerline. It comprises a carrier, identification, and two amplitude-modulated tones. The received amplitude of one tone is greater when the aircraft is to the left of the centerline, while that of the other is greater when the aircraft is to the right. The difference in depth of modulation (DDM) of the two tones is proportional to the course deviation when the aircraft is within 3° – 6° of the centerline. Beyond this, a constant reading is obtained.

The glideslope, transmitted in the 329–335-MHz band, provides the corresponding information in the pitch plane. The DDM is proportional to the path deviation when the elevation of the aircraft path is within a quarter of that recommended, typically 3° . Beyond that, it gives a constant reading, extending to at least 8° above the recommended path.

The marker beacons transmit constant tones at 75 MHz in an upward cone. The first beacon marks the start of the approach, the second is typically located 900m from the runway threshold, while a third is sometimes located at 300m. Marker beacons are often omitted where a DME beacon is colocated with the ILS.

9.4 Urban and Indoor Positioning

This section describes a number of relatively new terrestrial radio navigation techniques, designed to operate primarily in urban areas and indoors, where GNSS signals are subject to blockage, attenuation, and multipath interference. Techniques using mobile phones, signals of opportunity, GNSS repeaters, WLANs, UWB, and short-range beacons are covered [14].

9.4.1 Mobile Phones

The simplest mobile-phone positioning method is cell identification (ID), which just reports the base-station position. The resulting phone position error is up to 1 km in urban areas and 35 km in rural areas [15].

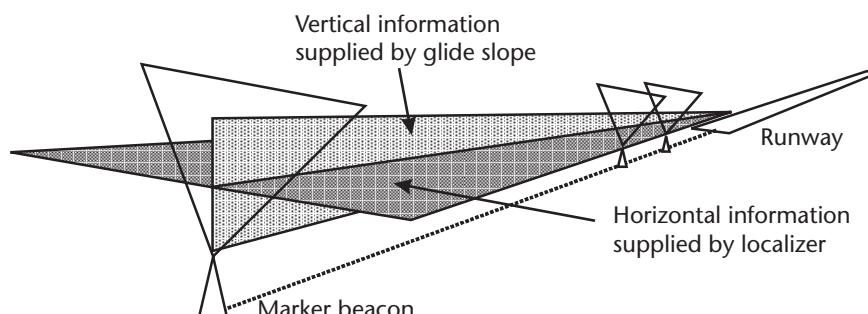


Figure 9.5 ILS schematic.

The North American CDMA and CDMA 2000 standards synchronize the base stations to UTC. Therefore phones receiving multiple base stations can obtain a position solution by passive ranging, which operates, using advanced forward-looking trilateration (AFLT), in the same manner as GNSS (see Section 6.1). With three base stations, there are two possible solutions for the latitude, longitude, and receiver clock offset, while a unique solution is obtained with four stations. Position accuracy varies from 20m to over 100m, depending on multipath [16].

The GSM and wideband CDMA (WCDMA) standards, used elsewhere, do not synchronize the base stations. To synchronize their communication frames, GSM phones perform a two-way ranging measurement with their current base station, accurate to about 150m, depending on multipath [15]. Alternatively, enhanced-observed time difference (EOTD) uses a network of receivers at known locations to measure the base station clock offsets in analogy with differential GNSS.

The Matrix method enables a position solution to be obtained using GSM or W-CDMA without modifying the network [17]. Instead, pseudo-range measurements from multiple phones are pooled. The pseudo-range from base station j to phone k is

$$\rho_{jk} = |\mathbf{r}_{ij}^i - \mathbf{r}_{ik}^i| + \delta\rho_{cj} - \delta\rho_{ck} + \delta\rho_{\epsilon jk} \quad (9.12)$$

where \mathbf{r}_{ij}^i and \mathbf{r}_{ik}^i are the phone and base station positions, $\delta\rho_{cj}$ and $\delta\rho_{ck}$ are the clock offsets, and $\delta\rho_{\epsilon jk}$ is the error. For M phones each using N base stations, there are MN measurements and $3M + N - 1$ unknowns, comprising $2M$ phone latitudes and longitudes and $M + N - 1$ relative clock offsets. The base station positions are known, while the phone heights are assumed to follow the terrain. Thus, a solution is available where $MN \geq 3M + N - 1$, which requires at least four base stations and three phones (or five stations and two phones). The base station clocks are stable over a few hours, so once they are calibrated single-phone position solutions may be obtained. Position accuracy is 50–100m for GSM and 25–50m for W-CDMA, depending on multipath.

Mobile phone signals may also be used to transmit GNSS differential corrections (Section 8.1) and assistance information (Section 8.3.3).

9.4.2 Signals of Opportunity

Signals of opportunity is the generic term for radio signals not designed specifically for navigation, such as radio and television broadcasts.

Carrier-phase positioning may be performed with any radio signals. Transmitter synchronization is not needed provided a base station is used. The methods described for GNSS in Section 8.2 are applicable generically; an example is the Cursor system [18]. Transmissions from at least three sites are needed to solve for latitude, longitude, and user-base station clock offset, while using more signals speeds up ambiguity resolution. Without using spread-spectrum modulation, there are no code measurements to limit the integer-ambiguity search space. However, the signal geometry changes more quickly with user motion for terrestrial transmit-

ters, while the ambiguity resolution can be aided using the received signal strength (RSS) and modulation [19].

The accuracy of carrier-phase positioning is about 2 percent of the wavelength with a good signal, while the range from the base station is determined by the coherence length of the carrier and the integer-ambiguity search space. In the medium frequency band, an accuracy of about 5m and a range of order 30 km is obtainable [18]. However, AM broadcasting is fast becoming obsolete in many countries.

With digital broadcasts, navigation receivers can track the known parts of the signal modulation. The Rosum Television Positioning System [20] uses the synchronization codes present in all digital television standards. It can also use synchronization signals found in analog television broadcasts. Base stations are still required, but the integer-ambiguity search space is much smaller than for carrier-phase positioning, so the range is greater. The accuracy is 5–20m, depending on multipath. However, in many cities, particularly within Europe, the broadcasters all share transmission sites, leading to inadequate signal geometry in some locations.

9.4.3 GNSS Repeaters

The GNSS signal strength within buildings may be increased by retransmitting the signal inside the building. However, the user equipment will report its position as that of the retransmission system's receiving antenna. If the retransmitted signals are cycled between four antennas within the building, the user location can be determined, using hyperbolic positioning, from the pseudo-range jumps each time the signal switches antenna, as these jumps are essentially time-difference measurements [21, 22]. The switching interval must be slow enough for the tracking loops to respond, limiting it to about 0.5 second, so user motion can degrade the accuracy. A faster switching cycle will require user-equipment modifications.

9.4.4 WLAN Positioning

Wireless local area network technology, also known as Wi-Fi and IEEE 802.11, provides computer networking at radio frequencies around 2.4 and 5 GHz. Base stations, known as access points (APs), are situated in offices and in public areas, such as cafés and airports. Each AP has a range of up to 100m, though attenuation by walls and buildings usually reduces this to a few tens of meters.

WLAN positioning uses the existing communications infrastructure, so is cheap to implement. A number of commercial positioning services are available. Some compute the position in the network server, others in the user equipment.

Each AP is identified by a unique code. Therefore, by simply identifying which APs are within range and using a database of their locations, such as Skyhook or Place Lab, a user position, accurate to a few tens of meters, can be obtained over a wide area [14].

WLAN transmissions are not time synchronized. Furthermore, in indoor and urban environments, the direct signal is often attenuated by walls, while reflected signals can be stronger. Therefore, timing-based positioning exhibits relatively poor accuracy [23]. Instead, more sophisticated WLAN positioning systems, such as the

Ekahau Positioning System and Microsoft Research Radar, measure the RSS from each AP in range. The RSS depends as much on building layout as distance from the transmitter. Therefore, the measured RSSs are compared with a database and the closest match within a region close to the previous known position is taken as the position fix. This is a feature-matching navigation technique (see Chapter 11). A position accuracy of 1–5m can be obtained [24, 25]. Databases are created using a mixture of measurements and signal propagation modeling.

RSSs can vary with time due to the opening and closing of doors and the movement of people, furniture, and equipment. Therefore, accuracy can be improved by using reference stations at known locations [26].

The Bluetooth, or IEEE 802.15.1, communication system also operates at 2.4 GHz. The maximum range is 100m, but a few tens of meters is more typical. However, reliable RSS measurements are difficult to obtain, while other proposed range measurement methods had yet to be tested in a multipath environment at the time of writing.

9.4.5 UWB Positioning

Ultra-wideband signals provide a solution to the severe multipath encountered in urban and indoor environments. They enable the multipath components of the received signal to be separated out, so the user need only track the direct signal, which arrives first. With a 1-GHz bandwidth, components separated by 0.6m may be resolved.

Dedicated wideband spectrum is not available. However, by employing spread-spectrum techniques (see Section 6.1.3), a very low PSD can be obtained, enabling UWB navigation signals to share spectrum with other users. In the United States, a PSD mask is defined, below which unlicensed transmissions are permitted. This has a maximum in the 3.1–10.6-GHz region used by many UWB systems. Similar regulations are being introduced elsewhere.

With the PSD limited by regulation, the range of a UWB signal depends on the ratio of the bandwidth to the data rate. Navigation does not require a high data rate, so ranges over a kilometer should be achievable. A multitude of UWB positioning systems were under development at the time of this writing. Three different types of signal are used: impulse radio (IR), frequency hopping (FH), and multicarrier (MC).

IR-UWB systems broadcast a series of subnanosecond pulses, which are inherently wideband. Range measurement and CDMA are obtained by amplitude modulating the pulse train by a ranging code, which is replicated within the receiver and correlated with the incoming signal [27]. A disadvantage of pulsed systems is that the ability to shape the signal spectrum to avoid certain frequencies is limited.

FH-UWB signals comprise GNSS-like BPSK signals, which hop frequency every few microseconds to achieve a wider bandwidth over the time constant of the code tracking loop (see Section 7.3.2). Interference between navigation signals is eliminated by ensuring that no two signals use the same frequency simultaneously [28].

MC-UWB signals comprise hundreds of regularly spaced carriers. CDMA is achieved using known initial phase offsets on each carrier. The received phase

difference between successive carriers, corrected for the initial offset, provides a TOA measurement [29].

For unambiguous range measurements, the transmitter range must not exceed the duty cycle. This is the code repetition interval for IR-UWB, the code- or hop-cycle repetition interval for FH-UWB and the inverse of the carrier separation for MC-UWB. For a 1-km range, a 3.3-ms duty cycle is needed, corresponding to a 300-kHz carrier spacing for MC-UWB; shorter-range systems can use wider spacing.

Depending on the degree of transmitter synchronization, positioning can make use of TOA, TDOA or two-way ranging measurements [30]. A submeter accuracy is obtainable in all cases.

9.4.6 Short-Range Beacons

Short-range beacons simply report their own location as the user passes them. The accuracy depends on how close the user approaches. Radio signposts, used for bus networks [31], and balises, used for rail navigation [32], are intended to be used alongside dead reckoning. A balise is a track-mounted transponder, powered by induction from a passing train. It can provide positioning to submeter accuracy.

Another option is to use radio-frequency identification (RFID) tags with a location database. The range varies from 3m for low-cost passive RFID tags, powered by RF induction, to 30m for active tags with their own power supplies [14].

MIT Cricket beacons broadcast their own positions at 418 MHz over a radius of a few meters, together with ultrasound pulses. By comparing the time of arrival of the radio and ultrasound signals, range may be measured to an accuracy of a few centimeters [14]. With a dense network of RFID tags or Cricket beacons every few meters, either system can provide stand-alone indoor positioning.

9.5 Relative Navigation

In relative navigation, ranges are measured between users, instead of between each user and a network of transmitters or transponders at known locations. It has the advantage of not requiring an infrastructure. However, it only provides the relative positions of the participants; to obtain absolute positions, the position of some participants must be determined by other means. There are two types of architecture: a chain and a network.

In a relative navigation chain, illustrated by Figure 9.6, each participant broadcasts a ranging signal on which its position and the uncertainty thereof are modulated. The participants at the beginning of the chain are at known locations. The others obtain their position using the signals from participants earlier in the chain. Thus, the further down the chain a participant is, the less accurate their position solution will be. To prevent positive feedback, a strict hierarchy must be maintained. However, this will change as the participants move.

An example of a chain system is the relative navigation (RelNav) function of the joint tactical information distribution system (JTIDS) and multifunctional information distribution system (MIDS), used by NATO aircraft, which communi-

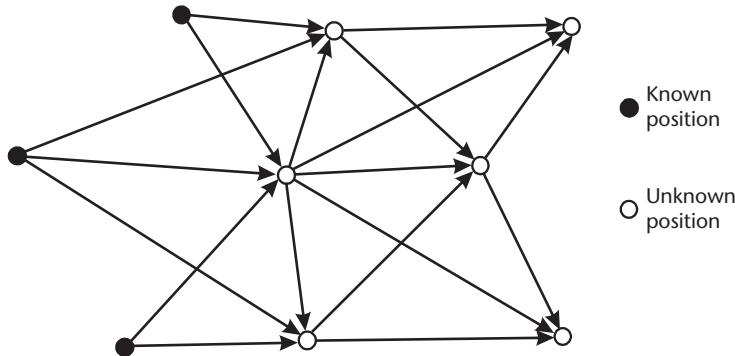


Figure 9.6 Relative navigation chain.

cate using Link 16 signals in the 960–1,215-MHz band [33, 34]. JTIDS/MIDS RelNav is designed to be integrated with inertial navigation, so each participant broadcasts a ranging signal every 3–12 seconds with a range of about 500 km. Participants are time synchronized so the ranges are obtained from TOA measurements. The position accuracy is 30–100m, depending on how far down the chain the user is.

In a relative navigation network, illustrated by Figure 9.7, ranging measurements in both directions are made between all pairs of users within range. These are relayed to a master station, which calculates the relative position of all of the participants [35]. At least four participants are needed for a two-dimensional solution and at least six for a three-dimensional solution. To obtain absolute positions for the network, only one participant need be in a known location. Alternatively, the measurements required to obtain an absolute position solution may be distributed between different members of the network.

An example of a relative navigation network is the position location reporting system (PLRS), used by the U.S. army and marines [33]. It uses the 420–450-MHz band, is accurate to 5–50m, partly depending on whether a user is moving, and can support up to 460 users over a 300-km square.

A version of relative navigation proposed for pedestrian navigation is measurement of the range between two foot-mounted INSs [36]. This calibrates both yaw-

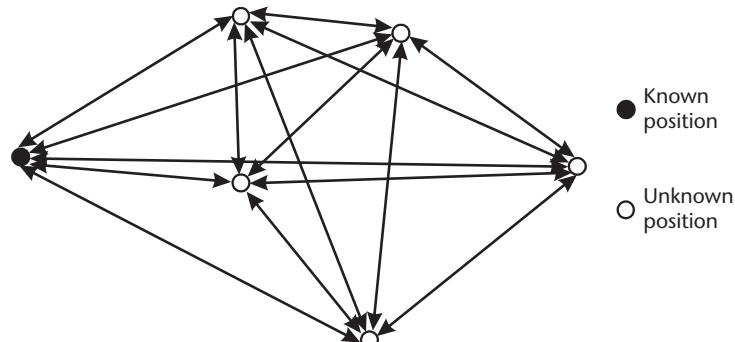


Figure 9.7 Relative navigation network.

axis gyro biases, complementing the calibration from regular ZVUs (see Section 13.3).

9.6 Tracking

Tracking differs from navigation in that the position of targets is determined within the network infrastructure. Tracking systems are outside the scope of this book, but they may be used for navigation simply by transmitting position data to the relevant users.

To track by radio, the users transmit while the base stations receive. Advantages of this approach are that more processing power may be available and phased arrays can be used to measure the AOA, which is not practical for most mobile receivers. The disadvantages are that the number of users and the update rate are limited.

Tracking may also be performed using radar surveillance, electro-optic sensors, and laser rangefinders.

9.7 Sonar Transponders

Radio navigation signals do not propagate underwater. Instead, submarines, remotely operated vehicles (ROVs), and autonomous underwater vehicles (AUVs) use sonar for underwater positioning. Long baseline (LBL) transponders are placed at known locations and two-way ranging used. A transducer aboard the host vehicle sends out a burst of sonar, known as a ping. The transponder then replies with a similar ping, a fixed interval after receiving the user's ping.

The speed of sound in water is around $1,500 \text{ m s}^{-1}$, so there can be an interval of several seconds between the sent and returned pings, during which time the motion of the host vehicle must be accounted for. Variation in the speed of sound, depending on temperature, depth, and salinity, can cause scale factor errors of a few percent, so this must be carefully measured.

A well-calibrated sonar ranging system operating at 8–12 kHz can achieve a range of up to 5 km and a ranging accuracy of 0.25–2m [37]. Higher frequency systems are more accurate, but have a shorter range. With a receiving array, the bearing of the transponder can also be measured to an accuracy between 0.05° and 5° , depending on frequency and array size [38].

References

- [1] Uttam, B. J., et al., “Terrestrial Radio Navigation Systems,” in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 99–177.
- [2] Enge, P., et al., “Terrestrial Radionavigation Technologies,” *Navigation: JION*, Vol. 42, No. 1, 1995, pp. 61–108.
- [3] Latham, R. W., and R. S. Townes, “DME Errors,” *Navigation: JION*, Vol. 22, No. 4, 1975, pp. 332–342.

- [4] Roth, G. L., and P. W. Schick, "New Loran Capabilities Enhance Performance of Hybridized GPS/Loran Receivers," *Navigation: JION*, Vol. 46, No. 4, 1999, pp. 249–260.
- [5] Last, J. D., "Enhanced Loran," *Proc. RIN Seminar on Non-GNSS Positioning*, London, U.K., October 2006.
- [6] Shmihluk, K., et al., "Enhanced LORAN Implementation and Evaluation for Timing and Frequency," *Proc. ION 61st AM*, Cambridge, MA, June 2005, pp. 379–385.
- [7] Roth, G. L., et al., "Performance of DSP—Loran/H-Field Antenna System and Implications for Complementing GPS," *Navigation: JION*, Vol. 49, No. 2, 2002, pp. 81–90.
- [8] Hartshorn, L., et al., "Performance of Loran-C 9th Pulse Modulation Techniques," *Proc. ION NTM*, Monterey, CA, January 2006, pp. 384–395.
- [9] Offermans, G. W. A., et al., "Integration Aspects of DGNSS and Loran-C for Land Applications," *Proc. ION 53rd AM*, Albuquerque, NM, June 1997.
- [10] Williams, P., and D. Last, "Mapping the ASFs of the North West European Loran-C System," *Journal of Navigation*, Vol. 53, No. 2, 2000, pp. 225–235.
- [11] Hartnett, R., G. Johnson, and P. Swaszek, "Navigating Using an ASF Grid for Harbor Entrance and Approach," *Proc. ION 60th AM*, Dayton, OH, June 2004, pp. 200–210.
- [12] Elias, A. L., "Aircraft Approach Guidance Using Relative Loran-C Navigation," *Navigation: JION*, Vol. 32, No. 1, 1985, pp. 1–15.
- [13] Vickers, D. B., et al., "Landing Systems," in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 597–641.
- [14] Kolodziej, K. W., and J. Hjelm, *Local Positioning Systems: LBS Applications and Services*, Boca Raton, FL: CRC Taylor and Francis, 2006.
- [15] Kitching, T. D., "GPS and Cellular Radio Measurement Integration," *Journal of Navigation*, Vol. 53, No. 3, 2000, pp. 451–463.
- [16] Kim, H. S., et al., "Performance Analysis of Position Location Methods Based on IS-801 Standard," *Proc. ION GPS 2000*, Salt Lake City, UT, September 2000, pp. 545–553.
- [17] Duffett-Smith, P. J., and P. Hansen, "Precise Time Transfer in a Mobile Radio Terminal," *Proc. ION NTM*, San Diego, CA, January 2005, pp. 1101–1106.
- [18] Duffett-Smith, P. J., and G. Woan, "The CURSOR Radio Navigation and Tracking System," *Journal of Navigation*, Vol. 45, No. 2, 1992, pp. 157–165.
- [19] Hall, T. D., C. C. Counselman III, and P. Misra, "Instantaneous Radiolocation Using AM Broadcast Signals," *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 93–99.
- [20] Rabinowitz, M., and J. J. Spilker, Jr., "A New Positioning System Using Television Synchronization Signals," *IEEE Trans. on Broadcasting*, Vol. 51, No. 1, 2005, pp. 51–61.
- [21] Caratori, J., et al., "UPGRADE RnS Indoor Positioning System in an Office Building," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1959–1969.
- [22] Jee, H.-I., S.-H. Choi, and S.-C. Bu, "Indoor Positioning Using TDOA Measurements from Switching GPS Repeater," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1970–1976.
- [23] Galler, S., et al., "Analysis and Practical Comparison of Wireless LAN and Ultra-Wideband Technologies for Advanced Localization," *Proc. IEEE/ION PLANS*, San Diego, CA, April 2006, pp. 198–203.
- [24] Eissfeller, B., et al., "Indoor Positioning Using Wireless LAN Radio Signals," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1936–1947.
- [25] Hatami, A., and K. Pahlavan, "A Comparative Performance Evaluation of RSS-Based Positioning Algorithms Used in WLAN Networks," *Proc. IEEE Wireless Communications and Networking Conference*, March 2005, pp. 2331–2337.
- [26] Chey, J., et al., "Indoor Positioning Using Wireless LAN Signal Propagation Model and Reference Points," *Proc. ION NTM*, San Diego, CA, January 2005, pp. 1107–1112.
- [27] Yu, H., "Long-Range High-Accuracy UWB Ranging for Precise Positioning," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 83–94.

- [28] Ingram, S., et al., "Ultra Wide Band Positioning as an Indoor Extension of GNSS," *Proc. GNSS 2003, ENC*, Graz, Austria, April 2003.
- [29] Cyganski, D., J. Orr, and W. R. Michalson, "Performance of a Precision Indoor Positioning System Using a Multi-Carrier Approach," *Proc. ION NTM*, San Diego, CA, January 2004, pp. 175–180.
- [30] Kang, D., et al., "A Simple Asynchronous UWB Position Location Algorithm Based on Single Round-Trip Transmission," *Proc. International Conference on Advanced Communication Technology*, February 2006, pp. 1458–1461.
- [31] El-Gelil, M. A., and A. El-Rabbany, "Where's My Bus? Radio Signposts, Dead Reckoning and GPS," *GPS World*, June 2004, pp. 68–72.
- [32] Mirabadi, A., F. Schmid, and N. Mort, "Multisensor Integration Methods in the Development of a Fault-Tolerant Train Navigation System," *Journal of Navigation*, Vol. 56, No. 3, 2003, pp. 385–398.
- [33] Fried, W. R., J. A. Kivett, and E. Westbrook, "Terrestrial Integrated Radio Communication-Navigation Systems," in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 283–312.
- [34] Ranger, J. F. O., "Principles of JTIDS Relative Navigation," *Journal of Navigation*, Vol. 49, No. 1, 1996, pp. 22–35.
- [35] Youssef, M., et al., "Self-Localization Techniques for Wireless Sensor Networks," *Proc. IEEE/ION PLANS*, San Diego, CA, April 2006, pp. 179–186.
- [36] Brand, T. J., and R. E. Phillips, "Foot-to-Foot Range Measurements as an Aid to Personal Navigation," *Proc. ION 59th AM*, Albuquerque, NM, June 2003, pp. 113–125.
- [37] Butler, B., and R. Verrall, "Precision Hybrid Inertial/Acoustic Navigation System for a Long-Range Autonomous Underwater Vehicle," *Navigation: JION*, Vol. 48, No. 1, 2001, pp. 1–12.
- [38] Jalving, B., and K. Gade, "Positioning Accuracy for the Hugin Detailed Seabed Mapping UUV," *Proc. IEEE Oceans '98*, 1998, pp. 108–112.

Dead Reckoning, Attitude, and Height Measurement

This chapter describes commonly used dead-reckoning (DR) techniques other than inertial navigation (Chapter 5). These measure the motion of the user with respect to the environment without the need for radio signals or extensive feature databases. However, an initial position solution must be supplied, as described in Section 5.5.1 for inertial navigation.

Magnetic field and pressure measurements may legitimately be classed as either dead reckoning or feature matching. They are described here alongside dead reckoning, as they are commonly used in the computation of a dead-reckoning navigation solution.

Section 10.1 describes attitude measurement, including the magnetic compass. Section 10.2 describes height and depth sensors. Sections 10.3 and 10.4 describe odometers and pedestrian dead reckoning, respectively. Section 10.5 describes Doppler radar and sonar. Finally, Section 10.6 discusses some other dead-reckoning techniques, including image processing and the ship's log.

10.1 Attitude Measurement

A number of attitude determination methods have been described in previous chapters. Inertial navigation (Chapter 5) determines attitude from angular rate measurements following a suitable initialization, while GNSS user equipment may measure attitude with multiple antennas, as described in Section 8.2.2. This section discusses roll and pitch measurement using accelerometers or tilt sensors, and then describes heading measurement using a magnetic compass. It concludes with a discussion of integrated heading measurement and the attitude and heading reference system.

10.1.1 Leveling

As described in Section 5.5.2, the roll and pitch attitude components of an inertial navigation solution are commonly initialized using leveling. The accelerometer triad is used to detect the direction of the acceleration due to gravity, which, neglecting local variations, denotes the down axis of the local navigation frame. The pitch and roll are given by (5.89):

$$\theta_{nb} = \arctan\left(\frac{-f_{ib,x}^b}{\sqrt{(f_{ib,y}^b)^2 + (f_{ib,z}^b)^2}}\right), \quad \phi_{nb} = \arctan 2(-f_{ib,y}^b, -f_{ib,z}^b)$$

where a four-quadrant arctangent function is used for roll. The same principle is used to calibrate the roll and pitch errors from the rate of change of the velocity error in INS/GNSS integration, transfer alignment, quasi-stationary alignment, zero velocity updates, and other INS integration algorithms (see Chapters 12 to 14).

In a navigation system without gyros, which are more expensive than accelerometers, leveling may be used as the sole means of determining the roll and pitch. However, (5.89) makes the assumption that the accelerometers are stationary, so only the reaction to gravity is measured. Thus, any acceleration disrupts the leveling process. For example, a 1 m s^{-2} forward acceleration will lead to a pitch determination error of about 100 mrad (5.7°).

Tilt sensors measure the roll and pitch attitude from the acceleration due to gravity directly. However, they are essentially accelerometers, so their attitude measurements are disrupted by acceleration in the same way as leveling measurements.

An alternative approach for air vehicles is to detect the horizon using a nose-mounted camera and image processing [1]. The noise is of order a degree, but the measurements are unaffected by acceleration.

10.1.2 Magnetic Heading

The Earth's geomagnetic field points from the magnetic north pole to the magnetic south pole through the Earth, taking the opposite path through the upper atmosphere, as illustrated by Figure 10.1. The field is thus vertical at the magnetic poles and horizontal near the equator. The magnetic poles slowly move over time, with

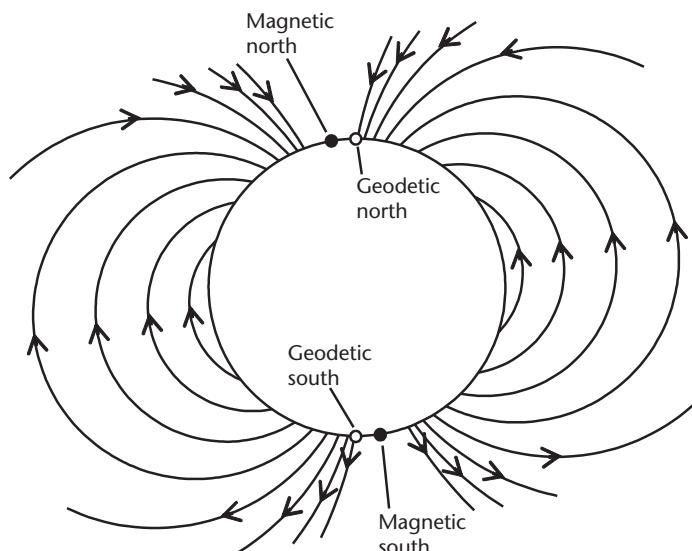


Figure 10.1 The Earth's geomagnetic field.

the north pole located in 2005 at latitude 82.7° , longitude -114.4° and the south pole at latitude -64.5° , longitude 137.9° , so the field is inclined at about 10° to the Earth's axis of rotation.

A magnetic field is described by the magnetic flux density vector, such that the force per unit length due to magnetic inductance is the vector product of the flux density and current vectors. The SI unit of magnetic flux density is the Tesla (T), where $1\text{T} = 1\text{NA}^{-1}\text{m}^{-1}$. The standard notation for it is \mathbf{B} . However, in the notation used here, this would be a matrix, while \mathbf{b} clashes with the instrument biases, so \mathbf{m} has been selected instead.

The flux density of the Earth's geomagnetic field, denoted by the subscript E , resolved about the axes of the local navigation frame, may be expressed using

$$\mathbf{m}_E^n(\mathbf{p}_b, t) = \begin{pmatrix} \cos \alpha_{nE}(\mathbf{p}_b, t) \cos \gamma_{nE}(\mathbf{p}_b, t) \\ \sin \alpha_{nE}(\mathbf{p}_b, t) \cos \gamma_{nE}(\mathbf{p}_b, t) \\ \sin \gamma_{nE}(\mathbf{p}_b, t) \end{pmatrix} B_E(\mathbf{p}_b, t) \quad (10.1)$$

where B_E is the magnitude of the flux density, α_{nE} is the declination angle or magnetic variation, and γ_{nE} is the inclination or dip angle of the Earth's magnetic field. All three parameters vary as functions of position and time.

The flux density varies from about $30\ \mu\text{T}$ at the equator to about $60\ \mu\text{T}$ at the poles, while the dip is essentially the magnetic latitude so is within about 10° of the geodetic latitude, L_b . The declination angle gives the bearing of the magnetic field from true north and is the only one of the three parameters needed to determine a user's heading from magnetic field measurements. It may be calculated as a function of position and time using global models, such as the 195-coefficient International Geomagnetic Reference Field (IGRF) [2] or the 248-coefficient U.S./U.K. World Magnetic Model (WMM) [3]. Regional variations, correlated over a few kilometers, occur due to local geology. Global models are typically accurate to about 0.5° , but can exhibit errors of several degrees in places [4]. Higher resolution national models are available for some countries. Short-term temporal variations in the Earth's magnetic field also occur due to magnetic storms caused by solar activity.

Magnetometers measure the total magnetic flux density, denoted by the subscript m , resolved about the axes of their body frame. Assuming that the magnetometer sensitive axes are aligned with those of any inertial sensors used, the body frame is denoted b . The magnetometers thus measure

$$\mathbf{m}_m^b = \mathbf{C}_n^b \begin{pmatrix} \cos \alpha_{nm} \cos \gamma_{nm} \\ \sin \alpha_{nm} \cos \gamma_{nm} \\ \sin \gamma_{nm} \end{pmatrix} B_m \quad (10.2)$$

where B_m , α_{nm} , and γ_{nm} are, respectively, the magnitude, declination, and dip of the total magnetic flux density. Applying (2.14),

$$\mathbf{m}_m^b = \begin{pmatrix} \cos \theta_{nb} & 0 & -\sin \theta_{nb} \\ \sin \phi_{nb} \sin \theta_{nb} & -\cos \phi_{nb} & \sin \phi_{nb} \cos \theta_{nb} \\ \cos \phi_{nb} \sin \theta_{nb} & \sin \phi_{nb} & \cos \phi_{nb} \cos \theta_{nb} \end{pmatrix} \begin{pmatrix} \cos \psi_{mb} \cos \gamma_{nm} \\ \sin \psi_{mb} \cos \gamma_{nm} \\ \sin \gamma_{nm} \end{pmatrix} B_m \quad (10.3)$$

where the magnetic heading, ψ_{mb} , is given by

$$\psi_{mb} = \psi_{nb} - \alpha_{nm} \quad (10.4)$$

Where the roll and pitch are zero, a magnetic heading measurement can be obtained from magnetometer measurements using

$$\tilde{\psi}_{mb} = \arctan2(-\tilde{m}_{m,y}^b, \tilde{m}_{m,x}^b) \quad (10.5)$$

whereas, where they are nonzero but known, the magnetic heading measurement is

$$\tilde{\psi}_{mb} = \arctan2\left(\frac{\tilde{m}_{m,y}^b - \cos \hat{\phi}_{nb} + \tilde{m}_{m,z}^b \sin \hat{\phi}_{nb}}{\tilde{m}_{m,x}^b \cos \hat{\theta}_{nb} + \tilde{m}_{m,y}^b \sin \hat{\phi}_{nb} \sin \hat{\theta}_{nb} + \tilde{m}_{m,z}^b \cos \hat{\phi}_{nb} \sin \hat{\theta}_{nb}}\right) \quad (10.6)$$

where a four-quadrant arctangent function should be used in both cases.

Floating-needle magnetic compasses have been used for centuries but do not provide an electronic readout. Electronic compasses use two or three orthogonally mounted magnetometers to measure the magnetic field and then calculate the magnetic heading using (10.5) or (10.6) as appropriate. The true heading is then given by

$$\tilde{\psi}_{nb} = \tilde{\psi}_{mb} + \alpha_{nE} \quad (10.7)$$

Types of magnetometer suitable for navigation systems are fluxgates, Hall-effect sensors, magnetoinductive sensors, and magnetoresistive sensors [4]. Magnetoinductive and magnetoresistive sensors are small and accurate to about $0.05 \mu\text{T}$ [5], which is good enough for most navigation applications, given the other error sources. Fluxgate sensors offer a better performance and can have dual sensitive axes, but are larger and more expensive, while the performance of Hall-effect sensors is much poorer.

A two-axis magnetic compass must be kept physically aligned with the horizontal plane to avoid errors in determining the heading. Where the small angle approximation applies to the roll and pitch, the heading error is

$$\delta\psi_{mb} \approx (\theta_{nb} \sin \psi_{mb} - \phi_{nb} \cos \psi_{mb}) \tan \gamma_{nm} \quad (10.8)$$

Thus, two-axis magnetic compasses are usually mounted in a set of gimbaled frames to keep them level. However, for road applications, changes in the magnitude

of the magnetometer measurements may be used to estimate the pitch and correct the heading accordingly [6].

A three-axis, or strapdown, magnetic compass uses an accelerometer triad to measure the roll and pitch using leveling (Section 10.1.1). However, this is disrupted by acceleration so is unsuited to high-dynamics applications. Acceleration-induced errors are also a problem for high-vibration applications, such as pedestrian navigation, but may be significantly reduced by smoothing measurements over the order of a second [7]. Where roll and pitch from an INS or AHRS are available, these should always be used in preference to leveling measurements.

Gimbaled and floating-needle compasses are also disrupted by acceleration and mitigate this using mechanical damping [8]. However, as with INS, they have largely been superseded by their strapdown counterparts.

A major problem for land applications is that magnetic fields are produced by man-made objects, such as vehicles, buildings, bridges, lampposts, and power lines [6, 7]. These can be significant several meters away and cannot easily be distinguished from the geomagnetic field. With a stand-alone magnetic compass, the only way of mitigating these local anomalies is to reject magnetometer measurements where the magnitude exceeds a certain threshold. However, this can still allow magnetic anomalies to produce heading errors of several degrees, while forcing the navigation system to rely on an out-of-date heading measurement when an anomaly is detected. Where the magnetic compass is integrated with another heading sensor (see Sections 10.1.3, 10.1.4, and 14.3.1.2), the integration process naturally smoothes out the effect of the local anomalies, while measurement innovation filtering (Section 15.3.1) can be used to reject the most corrupted magnetic compass measurements [7, 9].

The final obstacle to determining heading from magnetometer measurements is that, as well as the geomagnetic field and local anomalies, the magnetometers also measure the magnetic field of the navigation system itself, the host vehicle, and any equipment carried. This equipment magnetism is divided into hard-iron and soft-iron magnetism. Hard-iron magnetism is simply the magnetic fields produced by permanent magnets and electrical equipment. Soft-iron magnetism, however, is produced by materials that distort the underlying magnetic field. Soft-iron magnetism is large in ships, but small in aircraft and road vehicles.

The total magnetic flux density measured by a set of magnetometers is thus

$$\mathbf{m}_m^b = \mathbf{b}_m + (\mathbf{I}_3 + \mathbf{M}_m) \mathbf{C}_n^b (\mathbf{m}_E^n + \mathbf{m}_A^n) \quad (10.9)$$

where \mathbf{m}_E^n is the geomagnetic flux density as before, \mathbf{m}_A^n is the flux density from local magnetic anomalies, \mathbf{b}_m is the hard-iron flux density, resolved in the body frame, and \mathbf{M}_m is the soft-iron scale factor and cross-coupling matrix. Hard-iron and soft-iron magnetism are thus analogous to the biases, scale factor, and cross-coupling errors exhibited by inertial sensors (see Section 4.4).

The equipment and environmental magnetic flux densities may be distinguished by the fact that the equipment magnetism is referenced to the body frame, whereas the environmental magnetism is Earth-referenced. This enables \mathbf{b}_m and \mathbf{M}_m to be calibrated using a process known as *swinging*, whereby a series of measurements

are taken with the magnetic compass at different orientations, with the roll and pitch varied as well as the heading. This is done at a fixed location, so the environmental magnetic flux density may be assumed constant. The calibration parameters and environmental flux density may then be estimated using a nonlinear estimation algorithm, usually built into the magnetic compass. Following calibration, the magnetometer measurements are compensated using

$$\hat{\mathbf{m}}_m^b = (\mathbf{I}_3 + \hat{\mathbf{M}}_m)^{-1} \tilde{\mathbf{m}}_m^b - \hat{\mathbf{b}}_m \quad (10.10)$$

where $\hat{\mathbf{b}}_m$ and $\hat{\mathbf{M}}_m$ are the estimated hard and soft-iron magnetism. Where the magnetic compass is mounted in a large vehicle, a physical swinging process is not practical. Instead, some magnetic compasses perform electrical swinging using a self-generated magnetic field [8].

For applications where the magnetic compass is kept approximately level, a simpler four-coefficient calibration may be performed with the compass swung about the heading axis only. The correction is applied in the heading domain using [8]

$$\hat{\psi}_{mb} = \tilde{\psi}_{mb} + \hat{c}_{b1} \sin \tilde{\psi}_{mb} + \hat{c}_{b2} \cos \tilde{\psi}_{mb} + \hat{c}_{s1} \sin 2\tilde{\psi}_{mb} + \hat{c}_{s2} \cos 2\tilde{\psi}_{mb} \quad (10.11)$$

where \hat{c}_{b1} and \hat{c}_{b2} are the hard-iron calibration coefficients, and \hat{c}_{s1} and \hat{c}_{s2} the soft-iron coefficients. This calibration is only valid when the magnetic compass is level.

10.1.3 Integrated Heading Measurement

As discussed earlier, magnetic heading measurements are subject to errors induced by accelerations and local magnetic anomalies, so a more stable heading can be obtained by integrating the magnetic compass with a gyroscope (Sections 4.2 and 4.4) or differential odometer (Section 10.3). These sensors provide accurate measurements of short-term heading changes but are subject to long-term drift. Thus, they can be used to smooth out magnetic compass noise, while the magnetic compass calibrates the gyro/odometer drift.

The sensors may be integrated with a fixed-gain smoothing filter using

$$\hat{\psi}_{nb}(t) = W_m \hat{\psi}_{nb,m}(t) + (1 - W_m) [\hat{\psi}_{nb}(t - \tau) + \tilde{\omega}_{ib,z}^b \tau] \quad (10.12)$$

where $\hat{\psi}_{nb}$ is the integrated heading, $\hat{\psi}_{nb,m}$ the magnetic compass indicated true heading, $\tilde{\omega}_{ib,z}^b$ is the gyro or odometer-measured angular rate, and W_m is the magnetic compass weighting. However, it is better to use a Kalman filter (Chapter 3) as this enables the gyro or odometer bias to be calibrated, the sensor weighting optimized, and anomalous magnetic heading measurements filtered out. As discussed in Section 14.1, it is usually better to perform the heading integration in the integrated navigation system's central Kalman filter than in a dedicated local filter.

A single-axis gyroscope is only useful for measuring heading changes when the sensor is roughly level. Otherwise, the angular rate is underestimated by a factor of $1 - \cos \phi_{nb} \cos \theta_{nb}$. This error may be compensated where the pitch and roll are known, but is often neglected in land vehicles. A motorcycle undergoes large rolls when turning; however, the roll angle may be estimated as a function of the speed and yaw rate and used to compensate the gyro output [10].

A differential odometer overestimates heading changes on slopes (see Section 10.3).

10.1.4 Attitude and Heading Reference System

An attitude and heading reference system, or heading and attitude reference system (AHRS), comprises a low-cost IMU with automotive or tactical-grade sensors and a magnetic compass. It is typically used for low-cost aviation applications, such as private aircraft and UAVs, and provides a three-component inertial attitude solution without position and velocity. The attitude is computed by integrating the gyro measurements in the same way as in an INS (see Chapter 5), noting that the Earth-rotation and transport-rate terms in the local-navigation-frame implementation must be neglected where position and velocity are unknown.

The accelerometers measure the roll and pitch by leveling, as described in Section 10.1.1. This is used to correct the gyro-derived roll and pitch with a low gain to smooth out the corruption of the leveling measurements by host-vehicle maneuvers. The magnetic compass is used to correct the gyro-derived heading, again with a low gain used to smooth out short-term errors. The corrected gyro-indicated roll and pitch are used to determine the magnetic heading from three-axis magnetometer measurements using (10.6). Many AHRS incorporate maneuver detection algorithms to filter out accelerometer measurements during high dynamics. Using a Kalman filter to integrate data from the various sensors, as described in Section 14.3.1, enables the smoothing gains to be dynamically optimized and the gyro biases to be calibrated.

A typical AHRS provides roll and pitch to a 10-mrad accuracy and heading to a 20-mrad accuracy during straight and level flight, noting that performance depends on the quality of the inertial sensors and the type of processing used. Accuracy is typically degraded by a factor of 2 during high-dynamic maneuvers. More information on AHRS may be found in [8] while integration of AHRS with other navigation systems is discussed in Section 14.3.1.

Note that the term AHRS is sometimes used to describe a lower grade INS, rather than a device that only determines attitude.

10.2 Height and Depth Measurement

This section describes methods of determining absolute height and depth without using GNSS. The barometric altimeter, depth pressure sensor, and radar altimeter are covered.

10.2.1 Barometric Altimeter

A barometric altimeter uses a barometer to measure the ambient air pressure, p_b . Figure 10.2 shows how this varies with height. The height is then determined from a standard atmospheric model using [11, 12]

$$h_b = \frac{T_s}{k_T} \left[\left(\frac{p_b}{p_s} \right)^{-\frac{R k_T}{g_0}} - 1 \right] + h_s \quad (10.13)$$

where p_s and T_s are the surface pressure and temperature, h_s is the geodetic height at which they are measured, $R = 287.1 \text{ J kg}^{-1} \text{ K}^{-1}$ is the gas constant, $k_T = 6.5 \times 10^{-3} \text{ K m}^{-1}$ is the atmospheric temperature gradient, and $g_0 = 9.80665 \text{ m s}^{-2}$ is the average surface acceleration due to gravity. For differential barometry, the surface temperature and pressure are measured at a reference station and transmitted to the user. For stand-alone barometry, standard mean sea level values of $p_s = 101.325 \text{ kPa}$ and $T_s = 288.15 \text{ K}$ are assumed, in which case, $h_b - h_s$ is the orthometric height, H_b , defined in Section 2.3.3. Note that (10.13) only applies at orthometric heights up to 10.769 km. Above this, a constant air temperature of 218.15K is assumed.

The baro measurement resolution is about 10 Pa [9], corresponding to 1m of height near the surface and about 3m at an altitude of 10 km. The pressure measurement can also exhibit significant lags during rapid climbs and dives and is disrupted by turbulence and sonic booms. However, the main source of error in barometric height measurement arises from differences between the true and modeled atmospheric temperature and pressure. For stand-alone barometry, height errors can be several hundred meters. For differential barometry, the error increases with the distance from the reference station and the age of the calibration data. Rapid changes in the barometric height error can occur when the navigation system passes through a weather front.

Prior to the advent of GNSS, a baro was the only method of measuring absolute aircraft height at high altitudes, as the vertical channel of an INS is unstable (see

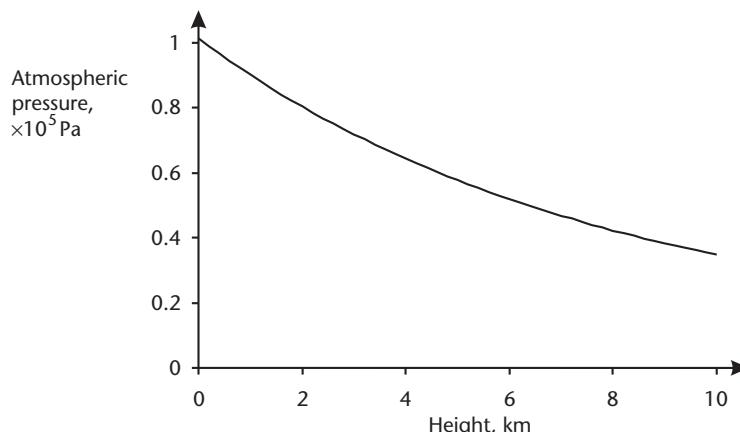


Figure 10.2 Variation of atmospheric pressure with height.

Section 5.6.2). To maintain safe aircraft separation, it is more important for different aircraft to agree on a height measurement than for that height to be correct. Therefore, at heights above 5.486 km, all aircraft use the standard mean-sea level values of p_s and T_s [13]. Furthermore, flight levels allocated by air traffic control are specified in terms of the barometric height, also known as pressure altitude, rather than the geodetic or orthometric height.

Aircraft baros have traditionally been integrated with the vertical channel of the INS using a third-order control loop. Using variable gains, the baro data can calibrate the INS during straight and level flights, when the baro is more stable, without the baro scale factor errors contaminating the INS during climbs and dives [14]. However, a baro-inertial loop presents a problem where the baro and INS are integrated with other navigation systems, such as GNSS, using a Kalman filter. This is because it is difficult to maintain an adequate model of the baro-inertial loop's behavior in the integration algorithm, particularly if the details are proprietary. Thus, the baro and INS should always be integrated as separate sensors. Baro integration is discussed in Section 14.3.2.

In recent years, barometric altimeters have been used in land applications. They are small and inexpensive and can be used in indoor pedestrian navigation to help determine which floor the user is on. Where GNSS signal availability is poor, such as in an urban canyon, use of a calibrated baro enables a position solution to be obtained with signals from only three GNSS satellites [15].

10.2.2 Depth Pressure Sensor

A depth pressure sensor determines the depth, d_b , of a submarine or AUV from a measurement of the water pressure, p_b , using

$$d_b = h_s - h_b = \frac{p_b - p_s}{\rho g} \quad (10.14)$$

where h_s and h_b are the geodetic heights of, respectively, the water surface and pressure sensor; p_s is the atmospheric pressure at the water surface; ρ is the water density; and g is the acceleration due to gravity. The water density varies as a function of the temperature and salinity but is approximately 10^3 kg m^{-3} for fresh water and $1.03 \times 10^3 \text{ kg m}^{-3}$ for seawater. The pressure increases by about 1 atmosphere (10^5 Pa) for every 10m of depth. Note that the surface height, h_s , may vary due to tidal motion.

10.2.3 Radar Altimeter

A radar altimeter (radalt) measures the height of an aircraft, missile, or UAV above the terrain by broadcasting a radio signal downward and measuring how long it takes the signal to return to the radalt after reflection off the ground below. The height above terrain is normally used directly as a landing aid, for ground collision avoidance, or for performing terrain-following flight. However, it may be combined with a terrain height database to determine the geodetic or orthometric height of the host vehicle where the latitude and longitude are known. A radalt and terrain

height database may also be used to perform terrain-referenced navigation as described in Section 11.1.

Radar altimeters generally transmit at 4.3 GHz, though some designs use 15.6 GHz. The range varies as the fourth root of the transmission power and is typically about 1,500m above the terrain. There are three main modulation techniques. A frequency-modulated continuous-wave (FMCW) radalt transmits a continuous signal at a varying frequency; the height above terrain is determined from the frequency difference between the transmitted and received signals. A pulsed radalt transmits a series of short pulses and determines the height from the time lag between the transmitted and received pulses. A spread-spectrum radalt operates in the same way as GNSS (see Section 6.1.3). A PRN code is modulated on the transmitted signal, and the received signal is then correlated with a time-shifted replica of the same code. The time shift that produces the correlation peak determines the height above terrain. All three types of radalt use a tracking loop to smooth out the noise from successive measurements and filter out anomalous returns [16].

The measurement accuracy of radalt hardware is about 1m. However, the accuracy with which the height above the terrain directly below the aircraft can be determined is only 1–3 percent. This is because the width of the transmitted radar beam is large, typically subtending $\pm 60^\circ$ in total with a full width at half maximum (FWHM) returned intensity of about 20° at 4.3 GHz. So, if the host vehicle is 1,000m above the terrain, the effective diameter of the radar footprint is about 350m. Thus, the terrain surrounding that directly below the aircraft has an effect on the height measurement. Where the terrain is flat, the return path length for the center of the footprint is shorter than that for the edge, so the radalt processing is biased in favor of the earliest part of the return signal. This is also useful for obstacle avoidance. Height measurement errors are larger where the aircraft is higher, as this gives a larger footprint, and where there is more variation in terrain height within the footprint. Figure 10.3 illustrates this. The beam width may be reduced by using a larger antenna aperture or a higher frequency.

When the host vehicle is not level, the peak of the transmitted radar beam will not strike the terrain directly below. However, mitigating factors are that the terrain reflectivity is usually higher at normal incidence, and the shortest return paths will still generally be from the terrain directly below. Thus, radalt measurements are valid for small pitch and roll angles but should be rejected once those angles exceed a certain threshold. Wider beam radalts are more tolerant of pitch and roll than narrower beam designs.

For ships, boats, submarines, and AUVs, sonar is similarly used to measure the height of the vessel above the seabed or riverbed.

10.3 Odometers

Odometers measure the rotation of the wheels of a land vehicle, giving the speed and distance traveled. Their earliest known use was in Roman chariots. Odometers have traditionally been fitted to the transmission shaft. However, most new vehicles have an odometer on each wheel. This is also known as a wheel speed sensor (WSS)

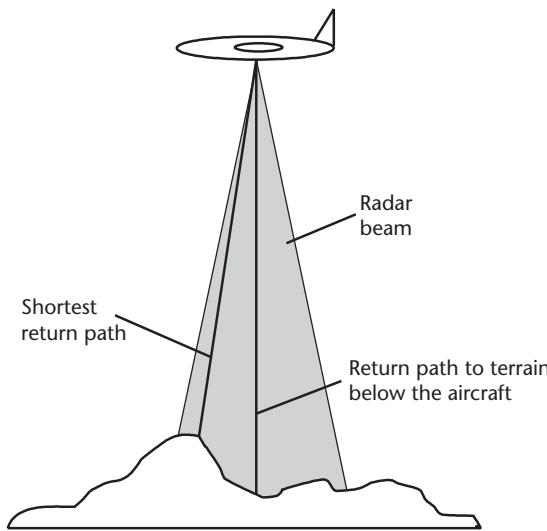


Figure 10.3 Effect of large footprint and terrain height variation on radar altimeter performance.

and is used for the ABS. By differentiating left and right odometer measurements, the yaw rate of the vehicle may be measured, a technique known as differential odometry. This was demonstrated by the Chinese in the third century CE with their *south-pointing chariot*.

To avoid mechanical wear, odometers use noncontact sensors. Most odometers mount a toothed ferrous wheel on the transmission shaft or wheel axle. As each tooth passes through a sensor, the magnetic flux density varies. Measuring this produces a pulsed signal, with the number of pulses proportional to the distance traveled. Differentiating this gives the speed. Low-cost odometers use passive sensors, based on variable reluctance. They exhibit poor signal-to-noise levels so are vulnerable to vibration and interference. They also do not work at speeds below about 1 m s^{-1} so are not recommended for navigation. Active sensors, often based on the Hall effect, give a strong signal at all speeds but are more expensive [6, 17]. Optical sensors may also be used but are vulnerable to dirt.

To describe navigation using odometers in a vehicle with front-wheel steering, it is useful to introduce three coordinate frames. The body frame, denoted b , describes the point on the host vehicle for which a navigation solution is sought. The rear-wheel frame, denoted r , is centered equidistant between the rear wheels along their axis of rotation and aligned with their direction of travel. This, in turn, is aligned with the body frame, so $\mathbf{C}_r^n = \mathbf{C}_b^n$. The front-wheel frame, denoted f , is centered equidistant between the front-wheel centers of rotation. It is aligned with the direction of travel of the front wheels, which is determined by the steering angle, ψ_{bf} . Thus,

$$\mathbf{C}_f^n = \mathbf{C}_b^n \begin{pmatrix} \cos \psi_{bf} & -\sin \psi_{bf} & 0 \\ \sin \psi_{bf} & \cos \psi_{bf} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (10.15)$$

The lever arms from the body frame to the rear and front-wheel frames are l_{br}^b and l_{bf}^b , respectively. Transmission-shaft measurements give the speed of the rear or front wheel frame, v_{er} or v_{ef} , depending on which are the driving wheels. Wheel speed measurements give the speed of each wheel, v_{erL} , v_{erR} , v_{efL} , and v_{efR} . The rear and front-wheel-frame speeds are then

$$v_{er} = \frac{1}{2} (v_{erL} + v_{erR}) \quad (10.16)$$

$$v_{ef} = \frac{1}{2} (v_{efL} + v_{efR})$$

When a road vehicle turns, each wheel travels at a different speed, while the front and rear wheels travel in different directions, moving along the forward (x) axis of the rear or front wheel frame as appropriate. Thus,

$$\mathbf{v}_{erL}^r = \begin{pmatrix} v_{erL} \\ 0 \\ 0 \end{pmatrix}, \mathbf{v}_{erR}^r = \begin{pmatrix} v_{erR} \\ 0 \\ 0 \end{pmatrix}, \mathbf{v}_{efL}^f = \begin{pmatrix} v_{efL} \\ 0 \\ 0 \end{pmatrix}, \mathbf{v}_{efR}^f = \begin{pmatrix} v_{efR} \\ 0 \\ 0 \end{pmatrix} \quad (10.17)$$

The velocity, in terms of both speed and direction, also varies across the vehicle body, as Figure 10.4 illustrates [18]. The rear track width, T_r , is the distance between the centers of the rear wheels' contact surfaces with the road. It is also the perpendicular distance between the tracks of those wheels along the road. The front track width, T_f , is the distance between the centers of the front wheels' contact surfaces. However, the perpendicular distance between the tracks is $T_f \cos \psi_{bf}$.

From (2.109) and (10.17), the body-frame velocity may be obtained from the rear or front wheel measurements using

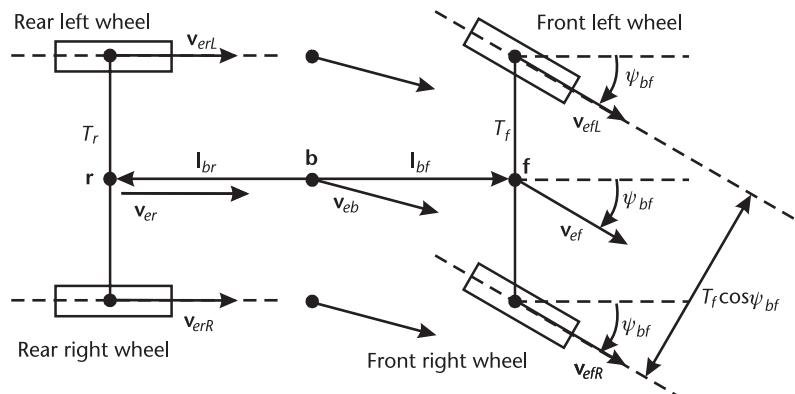


Figure 10.4 Road-vehicle wheel and body velocities during a turn.

$$\mathbf{v}_{eb}^b = \mathbf{C}_r^b \begin{pmatrix} v_{er} \\ 0 \\ 0 \end{pmatrix} - \boldsymbol{\omega}_{eb}^b \wedge \mathbf{l}_{br}^b = \begin{pmatrix} v_{er} \\ 0 \\ 0 \end{pmatrix} - \boldsymbol{\omega}_{eb}^b \wedge \mathbf{l}_{br}^b \quad (10.18)$$

or

$$\mathbf{v}_{eb}^b = \mathbf{C}_f^b \begin{pmatrix} v_{ef} \\ 0 \\ 0 \end{pmatrix} - \boldsymbol{\omega}_{eb}^b \wedge \mathbf{l}_{bf}^b = \begin{pmatrix} v_{ef} \cos \psi_{bf} \\ v_{ef} \sin \psi_{bf} \\ 0 \end{pmatrix} - \boldsymbol{\omega}_{eb}^b \wedge \mathbf{l}_{bf}^b \quad (10.19)$$

where the yaw component of $\boldsymbol{\omega}_{eb}^b$ may be obtained from differential odometry, as described later, or a yaw-axis gyro, correcting for the Earth rate. Neglecting the other components and the transport rate (Section 5.3.1), (10.18) and (10.19) simplify to

$$\begin{pmatrix} v_{eb,x}^b \\ v_{eb,y}^b \end{pmatrix} = \begin{pmatrix} v_{er} \\ 0 \end{pmatrix} + \begin{pmatrix} l_{br,y}^b \\ -l_{br,x}^b \end{pmatrix} \dot{\psi}_{nb} \quad (10.20)$$

or

$$\begin{pmatrix} v_{eb,x}^b \\ v_{eb,y}^b \end{pmatrix} = \begin{pmatrix} \cos \psi_{bf} \\ \sin \psi_{bf} \end{pmatrix} v_{ef} + \begin{pmatrix} l_{bf,y}^b \\ -l_{bf,x}^b \end{pmatrix} \dot{\psi}_{nb} \quad (10.21)$$

Neglecting vehicle roll and pitch, the change in position from time t to time $t + \tau_o$ is

$$\begin{pmatrix} \Delta r_{eb,N}^n(t, t + \tau_o) \\ \Delta r_{eb,E}^n(t, t + \tau_o) \end{pmatrix} = \int_t^{t + \tau_o} \begin{pmatrix} \cos \psi_{nb}(t') & -\sin \psi_{nb}(t') \\ \sin \psi_{nb}(t') & \cos \psi_{nb}(t') \end{pmatrix} \begin{pmatrix} v_{eb,x}^b(t') \\ v_{eb,y}^b(t') \end{pmatrix} dt' \quad (10.22)$$

Where the odometer(s) measure the average velocity from time t to $t + \tau_o$ and the heading rate and steering angle are both known, substituting (10.20) or (10.21) into (10.22) and integrating gives

$$\begin{pmatrix} \Delta r_{eb,N}^n(t, t + \tau_o) \\ \Delta r_{eb,E}^n(t, t + \tau_o) \end{pmatrix} \approx \begin{pmatrix} \cos \psi_{nb}(t) - \frac{1}{2} \dot{\psi}_{nb} \tau_o \sin \psi_{nb}(t) \\ \sin \psi_{nb}(t) + \frac{1}{2} \dot{\psi}_{nb} \tau_o \cos \psi_{nb}(t) \end{pmatrix} v_{er} \tau_o \quad (10.23)$$

$$+ \begin{pmatrix} \cos \psi_{nb}(t) & -\sin \psi_{nb}(t) \\ \sin \psi_{nb}(t) & \cos \psi_{nb}(t) \end{pmatrix} \begin{pmatrix} l_{br,y}^b \\ -l_{br,x}^b \end{pmatrix} \dot{\psi}_{nb} \tau_o$$

or

$$\begin{pmatrix} \Delta r_{eb,N}^n(t, t + \tau_o) \\ \Delta r_{eb,E}^n(t, t + \tau_o) \end{pmatrix} \approx \begin{pmatrix} \cos[\psi_{nb}(t) + \psi_{bf}(t)] - \frac{1}{2}(\dot{\psi}_{nb} + \dot{\psi}_{bf})\tau_o \sin[\psi_{nb}(t) + \psi_{bf}(t)] \\ \sin[\psi_{nb}(t) + \psi_{bf}(t)] + \frac{1}{2}(\dot{\psi}_{nb} + \dot{\psi}_{bf})\tau_o \cos[\psi_{nb}(t) + \psi_{bf}(t)] \end{pmatrix} v_{ef}\tau_o + \begin{pmatrix} \cos \psi_{nb}(t) & -\sin \psi_{nb}(t) \\ \sin \psi_{nb}(t) & \cos \psi_{nb}(t) \end{pmatrix} \begin{pmatrix} l_{bf,y}^b \\ -l_{bf,x}^b \end{pmatrix} \dot{\psi}_{nb}\tau_o \quad (10.24)$$

where the small angle approximation is applied to $\dot{\psi}_{nb}\tau_o$ and $\dot{\psi}_{bf}\tau_o$, and $\dot{\psi}_{nb}^2$ is neglected. Note that the steering angle and its rate of change are needed to navigate using front-wheel odometers. In either case, the latitude and longitude solutions are updated using

$$\begin{aligned} L_b(t + \tau_o) &= L_b(t) + \frac{\Delta r_{eb,N}^n(t, t + \tau_o)}{[R_N(L_b(t)) + h_b(t)]} \\ \lambda_b(t + \tau_o) &= \lambda_b(t) + \frac{\Delta r_{eb,E}^n(t, t + \tau_o)}{[[R_E(L_b(t)) + h_b(t)] \cos L_b(t)]} \end{aligned} \quad (10.25)$$

where R_N and R_E are given by (2.65) and (2.66).

Where individual wheel speed measurements are available, the yaw rate is

$$\dot{\psi}_{nb} = \frac{v_{erL} - v_{erR}}{T_r} \quad (10.26)$$

from the rear wheels or

$$\dot{\psi}_{nb} = \frac{v_{efL} - v_{efR}}{T_f \cos \psi_{bf}} - \dot{\psi}_{bf} \quad (10.27)$$

from the front wheels. Where odometer measurements are made over the interval t to $t + \tau_o$, the heading is updated using

$$\psi_{nb}(t + \tau_o) = \psi_{nb}(t) + \frac{1}{T_r} (v_{erL} - v_{erR}) \tau_o \quad (10.28)$$

or

$$\psi_{nb}(t + \tau_o) = \psi_{nb}(t) + \frac{2 + \dot{\psi}_{bf}\tau_o \tan \psi_{bf}}{2T_f \cos \psi_{bf}} (v_{efL} - v_{efR}) \tau_o - \dot{\psi}_{bf}\tau_o \quad (10.29)$$

Odometers measure the distance traveled over ground, not the distance traveled in the horizontal plane. Thus, if the host vehicle is traveling on a slope, the odometers

will overestimate the horizontal distance, as shown in Figure 10.5. For slopes of up to 140 mrad (8°), the error will be less than 1 percent. The yaw rate will also be overestimated. Where the roll and pitch are unknown, the position and heading errors must be corrected through integration with other navigation sensors. If the pitch is known, the odometer measurements may be corrected by multiplying v_{er} , v_{ef} , v_{erL} , v_{erR} , v_{efL} , and v_{efR} by $\cos \theta_{nb}$ in (10.23) to (10.29). Roll compensation is more complex, but land vehicles typically exhibit smaller rolls than pitches. Where there is no IMU, the pitch may be estimated from the rate of change of barometric height with distance traveled using

$$\hat{\theta}_{nb} = \arctan \left(\frac{\Delta h_b}{\sqrt{\Delta r_{eb,N}^n + \Delta r_{eb,E}^n}} \right) \quad (10.30)$$

The dominant error source in odometry is scale factor errors due to uncertainty in the wheel radii. Tire wear reduces the radius by up to 3 percent over the lifetime of a tire, while variations of order 1 percent can occur due to changes in pressure, temperature, load, and speed [19, 20]. Differential odometry is very sensitive to scale factor errors. For example, a 1 percent difference in scale factor error between the left and right wheels leads to a yaw-rate error of about 3° s^{-1} at a speed of 10 m s^{-1} . Thus, it is standard practice to calibrate the scale factor errors using other navigation sensors, such as GNSS, as described in Section 14.3.3.

Quantization resulting from the ferrous wheel teeth can be a significant source of short-term velocity and angular rate errors. However, as quantization errors are always corrected by subsequent measurements, the long-term position and heading errors are negligible [18]. A more significant source of random errors is road surface unevenness, particularly for differential measurements.

Odometers will produce false measurements of vehicle velocity where a wheel slips or skids due to rapid acceleration or braking on a slippery road. These can often be detected and filtered out using integrity-monitoring techniques, as described in Chapter 15. Odometers will also give misleading information when the vehicle is on a ferry, train, or trailer.

10.4 Pedestrian Dead Reckoning

Pedestrian navigation is one of the most challenging applications of navigation technology. A pedestrian navigation system must work in urban areas, under tree

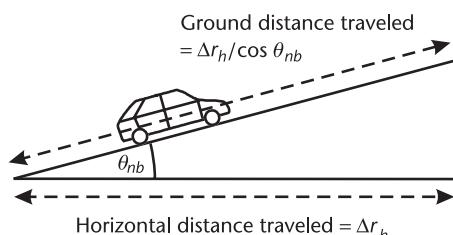


Figure 10.5 Effect of slope on measurement of distance traveled.

cover, and even indoors, where coverage of GNSS and most other radio navigation systems is poor. Inertial sensors are the only practical method of measuring forward motion by dead reckoning. However, for pedestrian use, they must be small, light, consume minimal power, and, in most cases, be low-cost. Thus MEMS sensors must be used. However, these provide very poor inertial navigation performance stand alone, while the combination of low dynamics and high vibration limits the calibration available from GNSS or other positioning systems. One solution, discussed in Section 13.3, is to use a shoe-mounted IMU and perform a zero velocity update on every step. The other solution, described here, is to use the inertial sensors for step counting, known as pedestrian dead reckoning. Note that a step is the movement of one foot with the other remaining stationary, while a stride is the successive movement of both feet. For body-mounted sensors, PDR gives significantly better performance than conventional inertial navigation, even when tactical-grade sensors are used [7].

Most PDR implementations use only accelerometers to sense motion. PDR can use a single accelerometer, mounted vertically on the body or along the forward axis of a shoe. However, using an accelerometer triad aids motion classification (see the following) and enables PDR to operate independently of the user's orientation [21].

A pedestrian dead-reckoning algorithm comprises three phases: step detection, step length estimation, and navigation-solution update. The step-detection phase identifies that a step has taken place. For shoe-mounted sensors, the measured specific force is constant when the foot is on the ground and variable when the foot is swinging, enabling steps to be easily identified [22]. For body-mounted sensors, the vertical or root sum of squares (RSS) accelerometer signals exhibit a double-peaked oscillatory pattern during walking, as Figure 10.6 shows. Steps can be detected from the “acceleration zero crossings” where the specific force rises above or drops below the acceleration due to gravity, with a recognition window used to limit false detections [21]. Alternatively, steps may be detected from the peaks in the accelerometer signals [23].

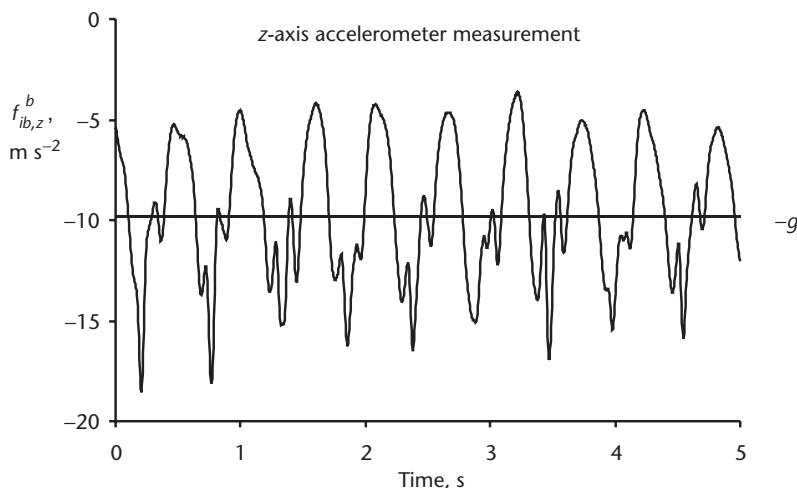


Figure 10.6 Vertical accelerometer signal during walking motion. (Data courtesy of QinetiQ Ltd.)

The length of a step varies not only between individuals, but also according to the slope and texture of the terrain, whether there are obstacles to be negotiated, whether an individual is tired, whether they are carrying things, and whether they are walking alone or with others. Thus, PDR implementations that assume a fixed step length for each user are only accurate to about 10 percent of distance traveled [24]. However, the step length varies approximately as a linear function of the step frequency [23]. It is also correlated with the variance of the accelerometer measurements [25] and the slope of the terrain [9] or vertical velocity [26]. By modeling the step length as a linear combination of a constant and terms dependent on these parameters, an accuracy of about 3 percent of distance traveled may be obtained [25, 26]. The model coefficients for each user may be estimated using measurements from GNSS or another positioning system, as discussed in Section 14.3.4.

PDR cannot be used for dead reckoning on its own, as it only measures the distance traveled, not the direction. It may be combined with a heading measurement (Section 10.1.3), noting that PDR may share the accelerometer triad of an AHRS (Section 10.1.4), in which case the position solution is updated using

$$\begin{aligned} L_b(+) &= L_b(-) + \frac{\Delta r_{PDR} \cos(\psi_{nb} + \psi_{bh})}{[R_N(L_b(-)) + h_b(-)]} \\ \lambda_b(+) &= \lambda_b(-) + \frac{\Delta r_{PDR} \sin(\psi_{nb} + \psi_{bh})}{\{[R_E(L_b(-)) + h_b(-)] \cos L_b(-)\}} \end{aligned} \quad (10.31)$$

where Δr_{PDR} is the PDR estimated step length, ψ_{bh} is the boresight angle, and the suffixes $(-)$ and $(+)$ denote before and after the update, respectively. The boresight angle is the angle in the horizontal plane between the forward axis of the sensor pack used for heading determination and the direction of motion. It is zero where the sensors are aligned with the direction of motion. Otherwise, it may be calibrated alongside the step-length-estimation coefficients [21].

Alternatively, PDR measurements may be used to calibrate the drift of an INS, sharing its accelerometers, as described in Section 14.3.4 [7, 27]. Where a tactical-grade IMU is used, this smoothes step-length estimation errors. However, there is little benefit in computing an inertial solution using automotive-grade sensors.

A basic PDR algorithm makes the assumption that all steps detected are forward walking, so backward and sideways steps lead to false measurements. A robust PDR implementation should thus incorporate a motion classification system so that different models are selected for different motion directions. This requires a full accelerometer triad. Different step length-estimation models may also be selected for running and for climbing stairs or steps. A hybrid PDR/inertial system may use the inertial velocity for motion classification.

10.5 Doppler Radar and Sonar

Where a radio or sound wave is transmitted to a receiver that is moving with respect to the transmitter, the receiver moves toward or away from the signal,

causing the wavefronts to arrive at the receiver at a faster or slower rate than they are transmitted at. Thus, the frequency of the received signal is shifted, to first order, by

$$\Delta f_{tr} \approx -\frac{f_t}{c} \mathbf{u}_{tr}^{\gamma T} \mathbf{v}_{tr}^{\gamma} \quad (10.32)$$

where f_t is the transmitted frequency, c is the speed of light or sound, \mathbf{u}_{tr}^{γ} is the line of sight unit vector from transmitter to receiver, and \mathbf{v}_{tr}^{γ} is the velocity of the receiver with respect to the transmitter. This is the Doppler effect. Where the transmitter and receiver are coincident on a body, b , but the signal is reflected off a surface, s , the Doppler shifts in each direction add, so

$$\Delta f_{tr} \approx -\frac{2f_t}{c} \mathbf{u}_{bs}^b T \mathbf{v}_{bs}^b \quad (10.33)$$

By reflecting three or more noncoplanar radio or sound beams off a surface and measuring the Doppler shifts, the velocity of the body with respect to that surface can be obtained. This is the principle of Doppler radar and sonar navigation. Most systems use a four-beam Janus configuration, as shown in Figure 10.7. The direction of each beam, indexed by i , with respect to the unit's body frame is given by a (negative) elevation angle, θ_{bsi} , and an azimuth, ψ_{bsi} , giving a line of sight vector of

$$\mathbf{u}_{bsi}^b = \begin{pmatrix} \cos \psi_{bsi} \cos \theta_{bsi} \\ \sin \psi_{bsi} \cos \theta_{bsi} \\ -\sin \theta_{bsi} \end{pmatrix} \quad (10.34)$$

The elevation is between -60° and -80° and is nominally the same for each beam. Nominal azimuths are either $30\text{--}45^\circ$, $135\text{--}150^\circ$, $210\text{--}225^\circ$, and $315\text{--}330^\circ$,

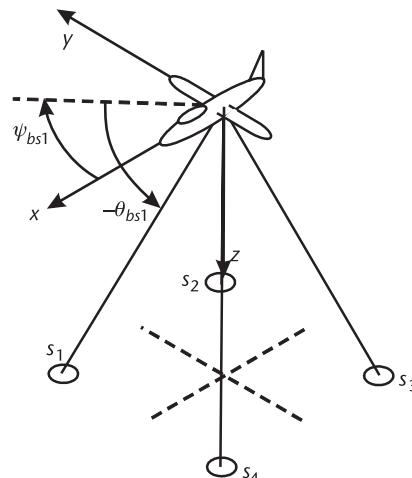


Figure 10.7 Typical four-beam Janus Doppler radar configuration.

as in Figure 10.7, or 0° , 90° , 180° , and 270° . The actual elevations and azimuths will vary due to manufacturing tolerances and may be calibrated and programmed into the unit's software.

The return signal to the Doppler unit comes from scattering of radar or sonar by objects in the beam footprint, not specular reflection off the surface. Thus the Doppler shift is a function of the relative velocity of the scatterers with respect to the host vehicle, not the range rate of the beams. Where the scatterers are fixed to the Earth's surface, $\mathbf{v}_{bsi}^b = -\mathbf{v}_{eb}^b$, so the Doppler unit measures velocity with respect to the Earth. The measurement model is thus

$$\begin{pmatrix} \tilde{\Delta f}_{tr1} \\ \tilde{\Delta f}_{tr2} \\ \tilde{\Delta f}_{tr3} \\ \tilde{\Delta f}_{tr4} \end{pmatrix} = \mathbf{H}\mathbf{v}_{eb}^b + \begin{pmatrix} w_{m1} \\ w_{m2} \\ w_{m3} \\ w_{m4} \end{pmatrix} \quad (10.35)$$

where w_{mi} is the measurement noise and the measurement matrix, \mathbf{H} , is

$$\mathbf{H} = \frac{2f_t}{c} \begin{pmatrix} \mathbf{u}_{bs1}^{b T} \\ \mathbf{u}_{bs2}^{b T} \\ \mathbf{u}_{bs3}^{b T} \\ \mathbf{u}_{bs4}^{b T} \end{pmatrix} \quad (10.36)$$

In analogy with the single-point GNSS navigation solution (Section 7.5.1), the Earth-referenced velocity is obtained by least-squares:

$$\hat{\mathbf{v}}_{eb}^b = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \begin{pmatrix} \tilde{\Delta f}_{tr1} \\ \tilde{\Delta f}_{tr2} \\ \tilde{\Delta f}_{tr3} \\ \tilde{\Delta f}_{tr4} \end{pmatrix} \quad (10.37)$$

To maintain a position solution, the attitude, \mathbf{C}_b^n , is required. Doppler radar and sonar do not measure attitude, so an AHRS or INS must be used. The latitude, longitude, and height are then updated using

$$\begin{pmatrix} L_b(+) \\ \lambda_b(+) \\ h_b(+) \end{pmatrix} = \begin{pmatrix} L_b(-) \\ \lambda_b(-) \\ h_b(-) \end{pmatrix} \quad (10.38)$$

$$+ \begin{pmatrix} 1/[R_N(L_b(-)) + h_b(-)] & 0 & 0 \\ 0 & 1/[R_E(L_b(-)) + h_b(-)] \cos L_b(-) & 0 \\ 0 & 0 & -1 \end{pmatrix} \hat{\mathbf{C}}_b^n \hat{\mathbf{v}}_{eb}^b$$

Noise arises because the Doppler shift varies across the footprint of each beam while the scatterers are distributed randomly. The noise standard deviation varies as the square root of the velocity [16]. Dynamic response lags, typically 0.1 second for radar, arise due to the use of frequency tracking loops in the receiver to smooth the noise. Velocity cross-coupling errors also arise due to residual beam misalignment and misalignment of the body frames of the Doppler unit and the attitude sensor.

Modern Doppler radars operate at 13.325 GHz and are usually frequency modulated. The technology was developed from the mid-1940s [28]. The typical accuracy of body-frame-resolved velocity over land is $0.06 \text{ m s}^{-1} \pm 0.2$ percent, though high-performance designs are about a factor of two better. Long-term position accuracy is about 1 percent of distance traveled with AHRS attitude and 0.15 percent with INS attitude [16]. The maximum altitude is at least 3000m above terrain [29].

Performance is poorer over water due to a large variation in the scattering coefficient with the angle of incidence. This causes the velocity to be underestimated by 1–5 percent, with the larger errors occurring over smooth water. Older Doppler radar systems provided the user with a land/sea calibration switch, reducing the residual errors to 0.3–0.6 percent (1σ). Newer designs typically use a modified beam shape, reducing the velocity errors to within 0.2 percent, while high-performance units measure the variation in scattering coefficient using additional or steerable beams [16]. In addition, the velocity is measured with respect to the water surface, not the Earth. Correction for this requires real-time calibration data or integration with a positioning system, such as GNSS.

Doppler radar is typically used for helicopter applications, where the slower speeds (below 100 m s^{-1}), compared to fixed-wing aircraft, lead to smaller velocity errors, while aircraft-grade INS are usually too expensive. Two-beam Doppler radar, omitting cross-track velocity, is sometimes used for rail applications.

A Doppler sonar system, also known as a Doppler velocity log (DVL), is used underwater to measure the velocity with respect to the bottom; it is applicable to the navigation of ships, submarines, ROVs, and AUVs. The range is a few hundred meters, with lower frequencies having a longer range, but producing noisier measurements. The speed of sound in water is about $1,500 \text{ m s}^{-1}$, but varies with temperature, depth, and salinity by a few percent. To get the best performance out of sonar, this must be correctly modeled.

Sonar is also subject to the effects of acoustic noise, while, in murky water, scattering of the sound by particles in the water above the bottom can introduce water-current-dependent errors. A well-calibrated and aligned Doppler sonar navigation system is accurate to about 0.5 percent of distance traveled [30, 31].

10.6 Other Dead-Reckoning Techniques

This section briefly reviews a number of other techniques that may be used to measure or calibrate velocity, resolved about the body frame. Image processing, landmark tracking, the correlation velocity log, air data, and the ship's log are

discussed. In each case, the velocity must be combined with an attitude measurement to update the position solution.

10.6.1 Image Processing

Velocity may be determined by comparing successive images of the terrain below an aircraft or the ground in front of a land vehicle, obtained using a fixed video or infrared camera [32, 33]. Images may be compared either by matching features or using optical flow, whereby the spatial derivative of the differences between the images is computed. To obtain a velocity measurement, the image plane must be transformed into the corresponding ground plane using the camera's height above terrain and line-of-sight angle of incidence. These are essentially fixed for a land vehicle. For an aircraft, a radar altimeter (Section 10.2.3) or a terrain height database, combined with the aircraft height solution, can provide height above terrain, while an INS or AHRS can provide attitude.

A laser scanner builds up a three-dimensional profile of the terrain below an aircraft, fully referenced to the body frame, enabling velocity to be obtained by correlating successive profiles [34]. Using fore and aft sensors increases the time interval between successive images or profiles of the same terrain, providing a higher velocity resolution for a given sensor resolution.

Use of forward-looking video cameras for land navigation has also been investigated. Challenges include distinguishing linear from angular motion and accounting for moving objects in the image [35]. The former may be mitigated by using a laser scanner to build a three-dimensional or range-bearing profile of the environment [36, 37].

10.6.2 Landmark Tracking

An electro-optic sensor, comprising a telescopic imaging sensor on a gimbaled mount, can provide precise measurements of the bearing and elevation of an unknown landmark from the host vehicle. Where the distance traveled and change in attitude are known, two successive measurements can be used to determine the location of the object relative to the host vehicle. The third and subsequent measurements can then be used to predict the distance traveled by the host vehicle if its change in attitude is known or the change in attitude where the distance traveled is known. Thus, although this technique cannot be used for stand-alone dead reckoning, it can aid calibration of the errors of an INS or other sensors [38, 39].

Using a laser scanner, the range of a landmark may also be measured, enabling velocity to be determined from landmark tracking alone, provided the host vehicle's attitude is known [40].

10.6.3 Correlation Velocity Log

A correlation velocity log (CVL) transmits a wide beam of sonar pulses straight down through water. The sonar is scattered by the bottom such that an interference pattern is produced. This is then measured by an array of receiving transducers

on the host vehicle. By correlating the interference patterns received from successive sonar pulses, an estimate of host vehicle velocity, resolved in the body frame, is obtained [41, 42]. A single-dimensional receiving array gives only forward velocity, while a two-dimensional array gives both horizontal components.

With a much wider beam, a CVL can operate at a much lower frequency than a Doppler velocity log with the same transducer size, giving a longer range. A CVL can operate at least 3,000m above the sea bed. Its velocity measurements are noisier than those of a DVL but are not affected by variations in the speed of sound. The long-term accuracy is similar at around 0.5 percent of distance traveled.

10.6.4 Air Data

Air speed is the forward component of an aircraft's velocity with respect to the air, as opposed to the ground. It is measured by differencing the pressure measured in a forward-pointing tube, known as a pitot, with that measured from a static port on the side of the aircraft [13]. It is accurate to about 2 m s^{-1} at speeds above 50 m s^{-1} but can be less accurate below this. Air speed is essential for flight control as the aircraft flies with respect to the air. However, it is a poor indicator of speed with respect to the ground, so is not generally used for navigation. Another navigation sensor, such as GNSS, can be used to calibrate the wind speed to within about 1 m s^{-1} [43].

10.6.5 Ship's Log

A ship's log measures the speed of a ship with respect to the water. Early designs used an actual wooden log, floating in the water. Speed was measured either by throwing the log off the front of the ship and timing how long it took the back of the ship to reach the log or by spooling out a knotted rope attached to a log in the water and timing the interval between knots. The first electronic logs were turbines, dragged behind the ship by a length of cable. The rate of rotation of the turbine was proportional to its speed through water. Now, sonar DVLs and CVLs are generally used. By measuring the sound scattered by particles suspended in water, they can measure velocity with respect to the water. These signals are distinguished from those returned from the bottom by timing. As with air data, the water current can be calibrated using other navigation sensors.

References

- [1] Winkler, S., et al., "Improving Low-Cost GPS/MEMS-Based INS Integration for Autonomous MAV Navigation by Visual Aiding," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1069–1075.
- [2] Macmillan, S., and S. Maus, "International Geomagnetic Reference Field—The Tenth Generation," *Earth, Planets and Space*, Vol. 57, No. 12, 2005, pp. 1135–1140.
- [3] McLean, S., et al., *The US/UK World Magnetic Model for 2005–2010*, Technical Report NESDIS/NGDC-1, Washington, D.C.: National Oceanic and Atmosphere Administration, and Edinburgh, U.K.: British Geological Survey, 2004.

- [4] Langley, R. B., "Getting Your Bearings: The Magnetic Compass and GPS," *GPS World*, September 2003, pp. 70–81.
- [5] Caruso, M. J., "Applications of Magnetic Sensors for Low Cost Compass Systems," *Proc. IEEE PLANS 2000*, San Diego, CA, March 2000, pp. 177–184.
- [6] Zhao, Y., *Vehicle Location and Navigation Systems*, Norwood, MA: Artech House, 1997.
- [7] Mather, C. J., P. D. Groves, and M. R. Carter, "A Man Motion Navigation System Using High Sensitivity GPS, MEMS IMU and Auxiliary Sensors," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 2704–2714.
- [8] Kayton, M., and W. G. Wing, "Attitude and Heading References," in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 426–448.
- [9] Ladetto, Q., et al., "Digital Magnetic Compass and Gyroscope for Dismounted Soldier Position & Navigation," *Proc. NATO RTO Symposium on Emerging Military Capabilities Enabled by Advances in Navigation Sensors*, Istanbul, Turkey, October 2002.
- [10] Coaplen, J. P., et al., "On Navigation Systems for Motorcycles: The Influence and Estimation of Roll Angle," *Journal of Navigation*, Vol. 58, No. 3, 2005, pp. 375–388.
- [11] *Manual of ICAO Standard Atmosphere*, Document 7488/2, Montreal, Canada: International Civil Aviation Organization, 1964.
- [12] Kubrak, D., C. Macabiau, and M. Monnerat, "Performance Analysis of MEMS Based Pedestrian Navigation Systems," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 2976–2986.
- [13] Osder, S. S., "Air-Data Systems," in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 393–425.
- [14] Ausman, J. S., "Baro-Inertial Loop for the USAF Standard RLG INU," *Navigation: JION*, Vol. 38, No. 2, 1991, pp. 205–220.
- [15] Käppi, J., and K. Alanen, "Pressure Altitude Enhanced AGNSS Hybrid Receiver for a Mobile Terminal," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 1991–1997.
- [16] Fried, W. R., H. Buell, and J. R. Hager, "Doppler and Altimeter Radars," in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 449–502.
- [17] Hay, C., "Turn, Turn, Turn: Wheel-Speed Dead Reckoning for Vehicle Navigation," *GPS World*, October 2005, pp. 37–42.
- [18] Carlson, C. R., J. C. Gerdes, and J. D. Powell, "Error Sources When Land Vehicle Dead Reckoning with Differential Wheelspeeds," *Navigation: JION*, Vol. 51, No. 1, 2004, pp. 13–27.
- [19] Bullock, J. B., et al., "Integration of GPS with Other Sensors and Network Assistance," In *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 459–558.
- [20] French, R. L., "Land Vehicle Navigation and Tracking," in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 275–301.
- [21] Käppi, J., J. Syrjärinne, and J. Saarinen, "MEMS-IMU Based Pedestrian Navigator for Handheld Devices," *Proc. ION GPS 2001*, Salt Lake City, UT, September 2001, pp. 1369–1373.
- [22] Cho, S. Y., et al., "A Personal Navigation System Using Low-Cost MEMS/GPS/Fluxgate," *Proc. ION 59th AM*, Albuquerque, NM, June 2003, pp. 122–127.
- [23] Judd, T., "A Personal Dead Reckoning Module," *Proc. ION GPS-97*, Kansas City, MO, September 1997, pp. 47–51.
- [24] Collin, J., O. Mezentsev, and G. Lachapelle, "Indoor Positioning System Using Accelerometry and High Accuracy Heading Sensors," *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 1164–1170.

- [25] Ladetto, Q., "On Foot Navigation: Continuous Step Calibration Using Both Complementary Recursive Prediction and Adaptive Kalman Filtering," *Proc. ION GPS 2000*, Salt Lake City, UT, September 2000, pp. 1735–1740.
- [26] Leppäkoski, H., et al., "Error Analysis of Step Length Estimation in Pedestrian Dead Reckoning," *Proc. ION GPS 2002*, Portland, OR, September 2002, pp. 1136–1142.
- [27] Soehren, W., and W. Hawkinson, "A Prototype Personal Navigation System," *Proc. IEEE/ION PLANS*, San Diego, CA, April 2006, pp. 539–546.
- [28] Tull, W. J., "The Early History of Airborne Doppler Systems," *Navigation: JION*, Vol. 43, No. 1, 1996, pp. 9–24.
- [29] Buell, H., "Doppler Radar Systems for Helicopters," *Navigation: JION*, Vol. 27, No. 2, 1980, pp. 124–131.
- [30] Jourdan, D. W., "Doppler Sonar Navigator Error Propagation and Correction," *Navigation: JION*, Vol. 32, No. 1, 1985, pp. 29–56.
- [31] Butler, B., and R. Verrall, "Precision Hybrid Inertial/Acoustic Navigation System for a Long-Range Autonomous Underwater Vehicle," *Navigation: JION*, Vol. 48, No. 1, 2001, pp. 1–12.
- [32] Hagen, E., and E. Heyerdahl, "Navigation by Images," *Modeling, Identification and Control*, Vol. 14, No. 3, 1999, pp. 133–142.
- [33] Sanchiz, J. M., and F. Pla, "Feature Correspondence and Motion Recovery in Vehicle Planar Navigation," *Pattern Recognition*, Vol. 32, 1999, pp. 1961–1977.
- [34] Vadlamani, A. K., and M. Uijt de Haag, "Use of Laser Range Scanners for Precise Navigation in Unknown Environments," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 1104–1114.
- [35] Giachetti, A., M. Campani, and V. Torre, "The Use of Optical Flow for Road Navigation," *IEEE Trans. on Robotics and Automation*, Vol. 14, No. 1, 1998, pp. 34–48.
- [36] Campbell, J. L., et al., "Flash-LADAR Inertial Navigator Aiding," *Proc. IEEE/ION PLANS*, San Diego, CA, April 2006, pp. 677–683.
- [37] Soloviev, A., D. Bates, and F. Van Graas, "Tight Coupling of Laser Scanner and Inertial Measurements for a Fully Autonomous Relative Navigation Solution," *Proc. ION NTM*, San Diego, CA, January 2007, pp. 1089–1103.
- [38] Hoshizaki, T., et al., "Performance of Integrated Electro-Optical Navigation Systems," *Navigation: JION*, Vol. 51, No. 2, 2004, pp. 101–121.
- [39] Pachter, M., A. Porter, and M. Polat, "INS-Aiding Using Bearings-Only Measurements of an Unknown Ground Object," *Navigation: JION*, Vol. 53, No. 1, 2006, pp. 1–19.
- [40] Hirokawa, R., et al., "Autonomous Vehicle Navigation with Carrier Phase DGPS and Laser-Scanner Augmentation," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1115–1123.
- [41] Grose, B. L., "The Application of the Correlation Sonar to Autonomous Underwater Vehicle Navigation," *Proc. IEEE Symposium on Autonomous Underwater Vehicle Technology*, Washington, D.C., June 1992, pp. 298–303.
- [42] Boltryk, P., et al., "Improvement of Velocity Estimate Resolution for a Correlation Velocity Log Using Surface Fitting Methods," *Proc. MTS/IEEE Oceans '02*, October 2002, pp. 1840–1848.
- [43] An, D., J. A. Rios, and D. Liccardo, "A UKF Based GPS/DR Positioning System for General Aviation," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 989–998.

Feature Matching

Feature-matching techniques determine the user's position by measuring features of the environment, such as terrain height or roads, and comparing them with a database in the same way that a person would compare landmarks with a map.

Feature-matching systems must be initialized with an approximate position solution in order to determine which region of the database to search. Limiting the database search area minimizes the computational load and the number of instances where there is more than one possible match between the measured features and those in the database. Most systems also require a velocity solution, usually from an INS or other dead-reckoning sensor, to determine the relative positions of the features they measure. Thus, feature matching is not a stand-alone navigation technique; it is only used as part of an integrated navigation system. Also, all feature-matching systems provide the occasional wrong fix, either because the database is out of date or due to selecting the wrong match where there are multiple possibilities. This must be handled by the integration algorithm (see Section 14.4.3).

Section 11.1 describes terrain-referenced navigation, Section 11.2 reviews image matching and Section 11.3 describes map-matching techniques. Finally, Section 11.4 discusses stellar navigation, gravity gradiometry, and magnetic field variations.

11.1 Terrain-Referenced Navigation

Terrain-referenced navigation determines position by comparing a series of terrain height measurements with a database. It is also known as terrain-aided navigation (TAN), terrain-contour navigation (TCN), and terrain contour matching (TCM), while the term *terrain-referenced navigation* is sometimes used to describe a broader family of terrain-based feature-matching techniques.

TRN development started in the 1960s. Systems currently available on the market include Terrain Profile Matching (TERPROM) [1] and Terrain Contour Matching (TERCOM) [2]. They are used for terrain collision avoidance as well as navigation.

Conventional TRN systems determine terrain height by using a radar altimeter (Section 10.2.3) to measure height above terrain and then subtracting this from the host vehicle's height solution. Figure 11.1 illustrates this. A number of terrain height databases, known as digital terrain models (DTMs) or digital elevation models (DEMs), of varying resolutions and coverage areas are available [3]. For

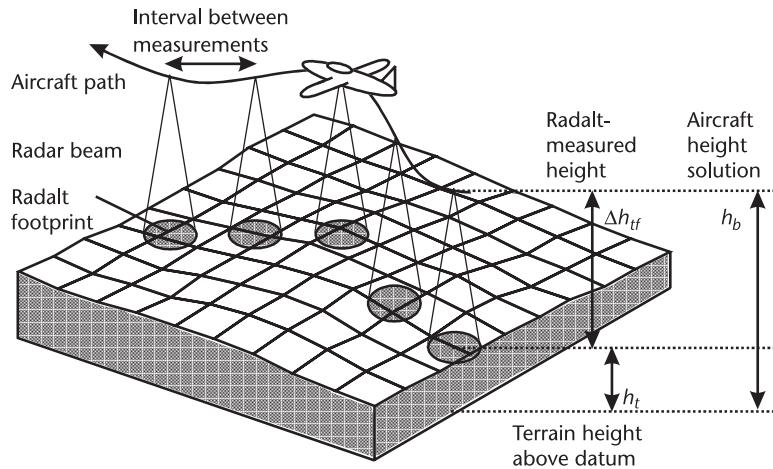


Figure 11.1 Terrain height measurement using a radar altimeter.

the optimum tradeoff between accuracy and data storage, the database resolution should match that of the radalt measurements. Most military TRN systems use level 1 Digital Terrain Elevation Data (DTED), collated by the NGA, which has a grid spacing of about 100m. About 2 GB of memory is needed for a whole Earth database [4]. Note that databases tend to use orthometric rather than geodetic height (see Section 2.3.3).

In theory, three terrain height measurements are needed to obtain a 3D position fix from TRN. However, as the measurement and database errors are often large compared to the short-distance terrain height variation, many more are usually needed for a unique match between the measurements and database. Over flat terrain and water, TRN cannot provide horizontal position fixes at all, only height information. The methods for obtaining position from the measurements may be grouped into two categories: sequential and batch processing. These are described next, followed by a discussion of performance and error sources. The section concludes by discussing TRN techniques using laser ranging for higher precision, barometric altimeters for land applications, and sonar for marine applications.

11.1.1 Sequential Processing

In sequentially processed TRN, used in TERPROM and Sandia Inertial TAN (SITAN) [5], an extended Kalman filter (Section 3.4.1) is used to estimate the position error of the host vehicle's navigation system. Each radar altimeter measurement, $\Delta\tilde{h}_{tf}$, is then processed separately to refine the position error estimate. The principal advantage of this method is a much lower processor load compared to batch-processing techniques.

The measurement innovation comprises the difference between the measured and database-indicated terrain height. Thus,

$$\delta z_T = \hat{h}_b - (0 \quad 0 \quad 1) \hat{\mathbf{C}}_b^n \mathbf{l}_{bf}^b - \Delta\tilde{h}_{tf} - b_{t,D}(\hat{L}_b, \hat{\lambda}_b) \quad (11.1)$$

where \hat{L}_b , $\hat{\lambda}_b$, and \hat{h}_b comprise the host navigation system's corrected latitude, longitude, and height solution; l_{bf}^b is the lever arm from the host-vehicle body frame to the radalt antenna; and $h_{t,D}$ is the geodetic height from the database.

Defining the state vector as

$$\mathbf{x}^n = \begin{pmatrix} \delta L_b \\ \delta \lambda_b \\ \delta h_b \\ \vdots \end{pmatrix} \quad (11.2)$$

where the latitude, longitude, and height errors, δL_b , $\delta \lambda_b$, and δh_b , are defined by (5.94), the measurement matrix is

$$\mathbf{H}_T = \begin{pmatrix} 1 & \frac{\partial h_{t,D}(\hat{L}_b, \hat{\lambda}_b)}{\partial L} & \frac{\partial h_{t,D}(\hat{L}_b, \hat{\lambda}_b)}{\partial \lambda} & 0 \end{pmatrix} \quad (11.3)$$

The key to successful operation of this method is correct determination of the terrain gradients, $\partial h_{t,D}/\partial L$ and $\partial h_{t,D}/\partial \lambda$ [6]. This is constrained by the database accuracy. However, the main limitation is that the gradient is calculated below the estimated host vehicle location, not the actual location. If the direction of the slopes at these locations differs, the estimated position can diverge from the truth instead of converging with it. In practice, sequential-processing TRN using an EKF does not work well where the combined position and database errors exceed a few hundred meters.

Two solutions to this problem have been developed. One is to replace the EKF with a nonlinear non-Gaussian estimation algorithm, such as a Viterbi algorithm, [7], point-mass filter [8], or particle filter [9, 10]. This removes the dependency on the terrain gradient at one point, but substantially increases the processor load. The other approach is to operate separate tracking and acquisition modes. The EKF is the tracking mode and operates when the position uncertainty is below its operational limit. The acquisition mode operates otherwise and may be a batch processing algorithm (see Section 11.1.2) or parallel filters, such as a MMAE filter bank (see Section 3.4.3) [11]. With a filter bank, each filter is offset in latitude and longitude and uses its own terrain gradient estimates. The filter that is closest to the truth exhibits the smallest measurement innovations over time.

11.1.2 Batch Processing

In batch-processed TRN, also known as template matching or terrain contour matching, a series of typically 5–16 terrain height measurements, known as a *transect*, are processed together. Each point of the transect is tagged with the host-vehicle navigation system's position solution, \hat{L}_b , $\hat{\lambda}_b$, and \hat{h}_b . Once all the data for a transect has been collected, it is processed in two stages. First, a probability distribution as a function of a position error, common to all points of the transect, is obtained by fitting the transect to the terrain height database. Second, a position correction is obtained from the distribution [12].

Obtaining a probability distribution as a continuous function of position is not practical, so a grid is used. The extent of the grid is typically matched to the 3σ error bounds of the host-vehicle navigation solution, while the spacing depends on the database resolution, terrain correlation length, and processing capacity. For an n -point transect, offset from the estimated host-vehicle position by ΔL , $\Delta \lambda$, and Δh , the measurement innovation vector is

$$\delta \mathbf{z}_T(\Delta L, \Delta \lambda, \Delta h) = \begin{pmatrix} \hat{h}_{b,1} + \Delta b - (0 \ 0 \ 1) \hat{\mathbf{C}}_{b,1}^n \mathbf{l}_{bf}^b - \Delta \tilde{h}_{tf,1} - h_{t,D}(\hat{L}_{b,1} + \Delta L, \hat{\lambda}_{b,1} + \Delta \lambda) \\ \hat{h}_{b,2} + \Delta b - (0 \ 0 \ 1) \hat{\mathbf{C}}_{b,2}^n \mathbf{l}_{bf}^b - \Delta \tilde{h}_{tf,2} - h_{t,D}(\hat{L}_{b,2} + \Delta L, \hat{\lambda}_{b,2} + \Delta \lambda) \\ \vdots \\ \hat{h}_{b,n} + \Delta b - (0 \ 0 \ 1) \hat{\mathbf{C}}_{b,n}^n \mathbf{l}_{bf}^b - \Delta \tilde{h}_{tf,n} - h_{t,D}(\hat{L}_{b,n} + \Delta L, \hat{\lambda}_{b,n} + \Delta \lambda) \end{pmatrix} \quad (11.4)$$

where the notation is as defined in the previous section. Using a mean square difference statistic, the likelihood of a particular offset, given the measurements, is [12]

$$\Lambda(\Delta L, \Delta \lambda, \Delta h) = \exp\left(-\frac{1}{2} \delta \mathbf{z}_T^T \mathbf{R}_T^{-1} \delta \mathbf{z}_T\right) \quad (11.5)$$

where the measurement noise covariance, \mathbf{R}_T , accounts for radalt measurement noise and database errors. Correlated measurement noise between different elements of the transect can occur as a result of radalt footprint errors (see Section 10.2.3) and host-vehicle velocity errors, so the matrix is not necessarily diagonal.

A three-dimensional likelihood grid can be impractical due to the processor load it imposes. The vertical component may be eliminated by differencing successive points in the transect [12]. Alternatively, the height offset for each latitude and longitude offset may be optimized by performing a least-squares fit, giving [13]

$$\Delta h(\Delta L, \Delta \lambda) = \frac{1}{n} \sum_{i=1}^n [\Delta \tilde{h}_{tf,i} + h_{t,D}(\hat{L}_{b,i} + \Delta L, \hat{\lambda}_{b,i} + \Delta \lambda) - \hat{h}_{b,i} + (0 \ 0 \ 1) \hat{\mathbf{C}}_{b,i}^n \mathbf{l}_{bf}^b] \quad (11.6)$$

Figure 11.2 shows a typical likelihood surface over a two-dimensional grid, where the measurement-database fit is ambiguous. There is more than one peak and significant levels of noise, making it difficult to determine the correct position offset to generate a fix from. The maximum likelihood point is not necessarily the correct fix due to noise. Alternatively, fitting a bivariate Gaussian distribution to the surface will overweight some parts of the solution space and downweight others. For example, the mean of the distribution can fall between the peaks where the likelihood is actually very low. A number of solutions to this problem of

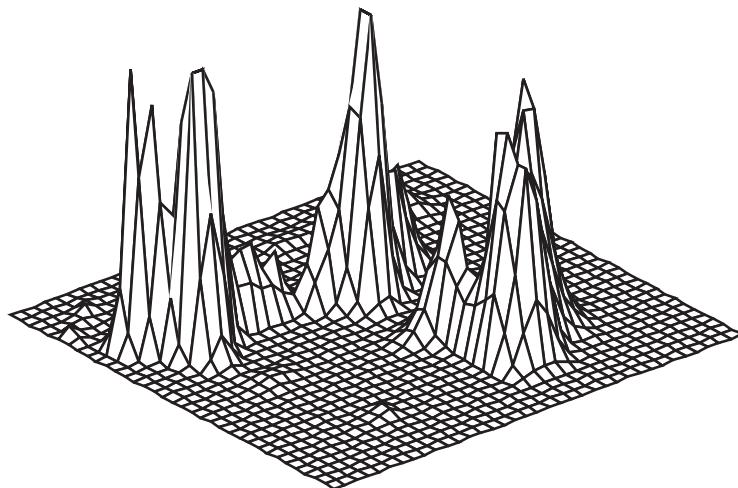


Figure 11.2 A TRN likelihood surface where the measurement-database fit is ambiguous. (After: [14].)

determining fixes from irregular likelihood surfaces have been developed; they are discussed in order of increasing processor load.

The simplest solution is to use transects sufficiently long that ambiguous fits rarely occur; at least 3 km is typically required [12]. However, long transects reduce the update rate, increasing the effect of host navigation system velocity error on the integrated position solution. Velocity errors can also smear the likelihood surface as they cause the best position correction for the end of the transect to diverge from that for the beginning.

The next simplest approach, known as the stockpot algorithm for robust TAN (SPARTAN), fits a Gaussian distribution to the likelihood surface but then adds the residuals of that fit to the likelihood surface from the next transect [12].

Another method is to fit multiple Gaussian distributions to the likelihood function. A simple way of doing this is to divide the likelihood grid into signal and noise cells and then fit a Gaussian function to each contiguous clump of signal cells [13]. A better performing, but processor-intensive, approach is to perform an iterative fit of a weighted sum of Gaussian functions to the likelihood surface. In either case, multiple position-fix hypotheses, each with an associated covariance and probability score, are passed to the integration algorithm (see Sections 3.4.4 and 14.4.3).

The most sophisticated batch-processing TRN algorithms modify the integration filter to accept the whole likelihood surface as a measurement. IGMAP [14] multiplies the integration algorithm's prior position error distribution by the TRN likelihood surface and then fits one or more Gaussian functions to the product, while Monte Carlo estimation algorithms, such as the particle filter or MCMCMC method [15], do away with Gaussian approximations altogether.

11.1.3 Performance

Under optimum conditions, conventional TRN operates with horizontal position errors of around 50m (1σ). Performance depends on the roughness of the terrain

below the host vehicle. Established TRN systems typically require terrain with an RMS gradient of at least 3 percent, and height varying in all directions, to operate. Greater roughness is needed to obtain the best performance. More sophisticated TRN algorithms offer better performance over low roughness terrain [10, 13] and will operate with a lower minimum roughness. Performance can also be limited where the terrain is too rough due to variation in terrain height within the radalt footprint, causing large errors in measuring the distance to the ground directly below the aircraft (see Section 10.2.3). Thus, to ensure continuous availability of TRN throughout a flight, the trajectory must be carefully selected [16].

A number of other factors affect TRN performance. The velocity accuracy of the inertial or dead-reckoning solution determines the accuracy of the assumed distance between successive radalt measurements and the degree to which TRN noise may be smoothed in the integrated navigation solution. The host-vehicle speed is important as the information available from TRN depends on the terrain correlation length; with a faster host vehicle, information is obtained at a faster rate. Performance degrades with increasing height above terrain, as a larger radalt footprint leads to noisier measurements. However, the fundamental limits to TRN performance are the database accuracy and the radalt beam width. Newer terrain height databases provide greater accuracy and resolution than DTED Level 1 [3], while interferometric radalts provide narrower beams than conventional units [4].

11.1.4 Laser TRN

The problem of a large sensor footprint can be eliminated by replacing the radalt with a laser rangefinder. With a conventional TRN algorithm, this reduces average position errors by up to 50 percent [17]. However, by scanning the laser from side to side, as shown in Figure 11.3, many more data points can be obtained for a given distance traveled [18, 19]. Using the host vehicle's integrated velocity and attitude solution, each data point is transformed from a time, range, and scanning angle to a relative position on the terrain from which the laser was reflected. The measurements can then be compared with the database using a batch-processing

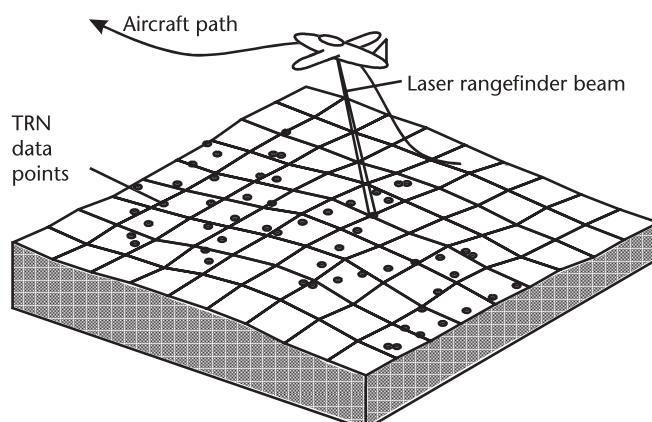


Figure 11.3 TRN data points with a scanning laser rangefinder.

algorithm (Section 11.1.2). With a DTED level 1 database, the ambiguity in the matches is largely removed, but the accuracy, integrated with an aircraft-grade INS, is only about 30m (1σ) [18].

To get the full benefit from using a laser scanner, a much higher database resolution is needed, with a 5m grid spacing and submeter accuracy. This enables a horizontal position accuracy within 10m (1σ) to be obtained [19]. With a 1m database grid spacing and integration with an aircraft-grade INS, an accuracy of about 2m (1σ) per axis is obtainable [20]. With a large number of data points, the processor load for conventional batch processing is very high. Solutions include subsampling the laser data to match the terrain correlation length and gradient-based matching algorithms. A further issue is that current laser rangefinders have shorter ranges than radalts and are sensitive to the weather.

11.1.5 Barometric TRN

For land vehicles and pedestrians, the navigation system generally maintains a constant height above terrain, so a barometric altimeter (Section 10.2.1), with the bias calibrated (Section 14.3.2), may be used to measure terrain height. Thus, in principle, terrain-referenced navigation may be performed [21]. However, as the host vehicle or user speed is much lower, a high-resolution database is needed to capture sufficient terrain height variation. Further research is needed to determine the range of terrain over which such a system would operate.

11.1.6 Sonar TRN

Submarines, AUVs, and ROVs can use a multibeam sonar echo-sounder to measure the range from the vehicle to a number of points on the sea bed or river bed. The relative positions of these points, obtained from the sonar measurements, is known as a *bathymetric profile*. TRN may then be performed by matching this with a suitable database. As with other forms of TRN, the accuracy depends on the terrain height variation but is of the order of 100m [22, 23].

11.2 Image Matching

Image-matching navigation systems obtain position fixes by comparing images of the terrain below an air vehicle with a database. They may also be used for dead reckoning by comparing successive images (see Section 10.6.1), though this tends to be less accurate.

Images may be captured using a standard commercial video or digital stills camera, while infrared cameras work well at night and are less susceptible to the weather. The image must be correctly scaled in terms of the distance along the ground that each pixel corresponds to. This requires the height of the camera above the terrain, which may be obtained using a radalt, by differencing the host-vehicle height with a terrain height database, or by comparing the image velocity with that of the host vehicle. The line-of-sight angle is also required and is obtained from the host-vehicle attitude solution. Images from a downward-looking camera

are easier to process than those from a forward- or sideways-looking camera, as a constant scaling may be assumed for the whole image and there is very little shadowing of features by the terrain. However, downward-looking cameras tend to cover a smaller area of terrain, providing fewer features to match with the database.

Use of an active sensor, such as a laser scanner [18] or SAR [24], provides a range measurement for every image pixel. It also enables contour TRN and image matching to be performed using the same sensor. This reduces hardware costs, removes relative alignment errors, and enables range data to be used to aid detection of image boundaries [18].

Like TRN, image matching does not work over all terrain, as suitable distinct features, generally man-made, are needed for matching. However, more features tend to be found on flat terrain, over which TRN performance is poor, so the two techniques are complementary. Image matching does not work over water.

Systematic error sources in image matching include database errors, sensor alignment errors, camera lens distortions, and uncompensated scaling variations. Noise-like errors arise from focusing errors and the resolution limits of the sensor, database, and image-matching algorithms. The statistical parameters assumed in the image-matching algorithms must be tuned to account for these errors. The remainder of this section describes two image-matching navigation techniques, scene matching by area correlation (SMAC) and continuous visual navigation (CVN).

11.2.1 Scene Matching by Area Correlation

SMAC operates by correlating the captured image, following appropriate scaling and rotation, with a corresponding image from the database. Comparing full color, or even grayscale, images is not effective due to seasonal variations, changes in lighting conditions, and day/night contrast inversion of infrared images. Instead, both the captured and stored images are processed to extract boundaries, as illustrated by Figure 11.4 [25, 26]. This also has the advantage of reducing the correlation processing load and database storage requirements.

One correlation method varies the latitude and longitude offsets between the two images, counting the number of boundary pixels that match, allowing a certain

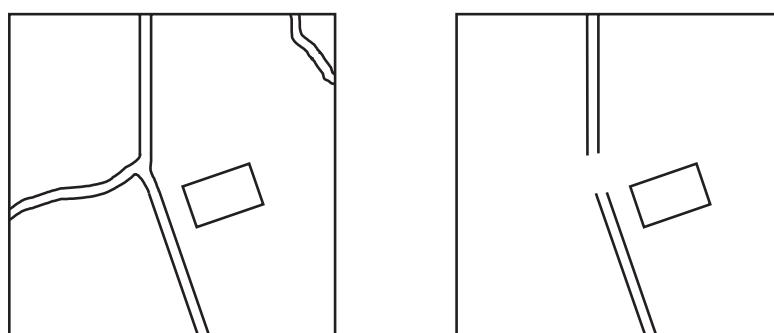


Figure 11.4 Example of boundary features (left) and line features (right) extracted from an image.

leeway for noise. The offset with the most matching pixels provides the position fix. As with batch-processed TRN, the offset search area should correspond to about the 3σ uncertainty bounds of the host-vehicle position solution.

Even binary images require a lot of storage, so SMAC systems do not store a complete database. Instead, a series of landmark images are stored, from which unambiguous position fixes may be made. Historically, storage limitations have led to the need to preplan the host-vehicle trajectory, limiting the range of applications for which SMAC is suited. SMAC position fixes are accurate to a few tens of meters, but the overall accuracy also depends on the INS or dead-reckoning system used to navigate between landmarks.

11.2.2 Continuous Visual Navigation

CVN [26, 27] was designed to overcome the limitations of SMAC by providing continuous position fixing without the need for prior route planning. This requires the database to cover the whole region of operation. To reduce the amount of storage required for a given area of terrain, CVN matches only straight-line features. Sources include roads, rail tracks, buildings, and field boundaries. Line-feature storage is very efficient, as only the start and end points are needed.

Like SMAC, CVN first processes the scaled and rotated images to extract boundaries. Line features are then extracted from the boundary images, as Figure 11.4 shows. Each image line feature is then compared with those lines in the database that are of a similar direction and within the region bounded by about the 3σ uncertainty of the host-vehicle position solution. The offset between the image and database lines then provides a measurement of the position error in the perpendicular horizontal direction to the lines.

To obtain a position fix from a single image, orthogonal line features are needed. To enable images with single or parallel line features to contribute to the navigation solution, making maximum use of the available data, CVN uses a Kalman filter to estimate the errors in the host-vehicle navigation solution. With the host-vehicle velocity solution providing the distance traveled between images, line-fix measurements from different images may be combined to produce a position fix (see Section 14.4.2).

A single database line may often be matchable with multiple image lines or vice versa, leading to ambiguous fixes. Any feature-matching system produces the occasional wrong fix. In CVN, this is resolved using multiple hypothesis filtering, as described in Sections 3.4.4 and 14.4.3, enabling consistency to be checked across successive fixes.

Over suitable terrain, CVN exhibits a horizontal radial position accuracy of about 10m when integrated with an aviation-grade INS and 20m when integrated with a tactical-grade INS [27].

11.3 Map Matching

Road vehicles usually travel on roads, trains always travel on rails, and pedestrians don't walk through walls. Map matching, also known as map aiding or snap to

map, is used in land applications to correct the integrated navigation solution by applying these constraints.

Map matching is most commonly used in road vehicle navigation, generally integrated with GNSS and dead reckoning [28]. The map-matching algorithm compares the position solution from the navigation system with the roads in its database and supplies a correction perpendicular to the road direction if the navigation solution drifts off the road. Figure 11.5 illustrates this. While the host vehicle is traveling in a straight line, map matching can only provide one-dimensional positioning. Turns are needed to obtain a two-dimensional fix. However, in urban canyons, where GNSS satellite visibility is poor, map matching's cross-track positioning complements the along-track positioning that GNSS may be limited to.

The key to successful map matching is identification of which road segment the vehicle is on. The simplest techniques, known as *point-to-point* or *point-to-curve* matching, just find the nearest road in the database to the navigation system's reported position. This works well in rural areas where the road separation is much greater than the uncertainty bounds of the navigation solution. However, in urban areas, where the road network is denser and GNSS performance can be poor, this can often produce errors, as Figure 11.6 illustrates.

Road segment identification can be improved by also matching the direction of travel. However, this can still lead to a false match, particularly where the roads are arranged in a grid pattern, as shown in Figure 11.6. Traffic-rule information,

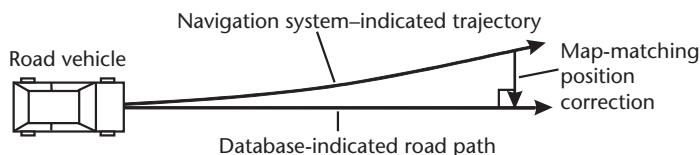


Figure 11.5 Map-matching position correction for a road vehicle.

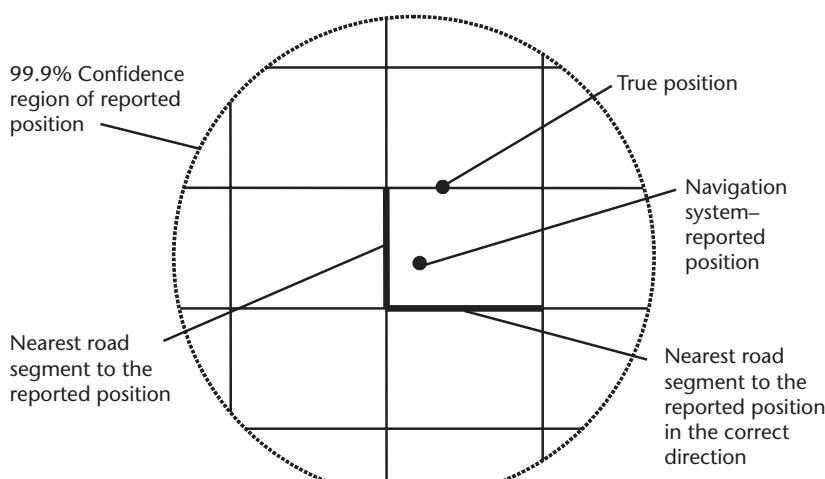


Figure 11.6 False road identification in an urban area.

such as one-way streets and illegal turns can also help. However, reliable road-segment identification requires *curve-to-curve* matching techniques, which match a series of consecutive reported positions to the map database [29]. This may be done using transect-matching techniques analogous to those used for batch-processed TRN (Section 11.1.2) or by maintaining multiple hypotheses and gradually eliminating those hypotheses where successive position fixes cannot be connected by a road segment [30].

Map matching must also cope with the vehicle not being on any of the roads in the database. This will happen when parking or driving along a new road, in which case it is better to produce no map-matching measurements than a wrong fix [31].

A limit to map-matching accuracy is that it is not practical to identify which lane the vehicle is in where there is more than one lane per direction, though divided highways are usually represented in map databases as pairs of one-way streets, rather than single roads. A navigation system incorporating map matching can achieve a horizontal position accuracy of about 10m (1σ) [32].

For rail applications, map matching can be used to detect when a train moves between parallel tracks by matching changes in its heading to maps of points stored in the database. Away from points, track maps can be used to maintain heading calibration [33].

For pedestrian navigation, the user position is nowhere near as constrained. However, there is still a role for map matching. Figure 11.7 illustrates some examples. If the navigation solution and database indicate the user is walking perpendicularly through a wall, the position solution needs to be shifted to the nearest doorway, while if the user appears to be drifting through a wall to their side, the direction of travel needs correcting. Detection of stair climbing can also be used to trigger a position correction.

11.4 Other Feature-Matching Techniques

This section discusses the use of feature matching for navigation using other types of sensor information. Stellar navigation, gravity gradiometry, and magnetic field

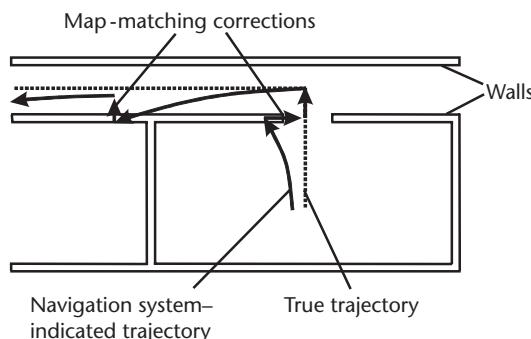


Figure 11.7 Application of map matching to pedestrian navigation.

variations are covered in turn. Note that WLAN positioning is described in Section 9.4.4.

11.4.1 Stellar Navigation

Unlike other feature-matching techniques, stellar, or celestial, navigation measures attitude rather than position. The line-of-sight vectors of the stars in the CIRS inertial frame may be determined from a star catalog. By measuring the azimuth and elevation of at least two stars, assuming the orientation of the sensor with respect to the body frame is known, the attitude of the navigation system with respect to the inertial frame, C_b^i , may be determined.

Two types of sensors may be used. A conventional star tracker comprises a telescope with a light detector mounted on gimbals [34]. It is pointed in the expected direction of a known star and then scanned to maximize the light detector signal. Thus, an approximate attitude solution from an INS or AHRS is required. About three star fixes per minute can be obtained. Details of only the 50–100 brightest stars need be stored in the database.

A star imager, or strapdown star tracker, comprises a charge-coupled-device (CCD) camera fixed to the host vehicle body. The resolution is about $1,000 \times 1,000$ pixels, while the field of view varies from about 2° to 40° . Attitude is determined by pattern matching the image with a database of about 10,000 stars [35]. A star imager may be used as the sole attitude sensor.

Both types of sensor provide attitude measurements accurate to about $5 \mu\text{rad}$ ($3 \times 10^{-4}^\circ$) in space and $50 \mu\text{rad}$ ($3 \times 10^{-3}^\circ$) in the atmosphere. Star trackers and imagers work in daylight as well as at night, while infrared sensors are less sensitive to cloud cover than those operating at visible wavelengths.

Integrating stellar navigation with an aircraft-grade INS essentially calibrates out the gyro drift, enabling the horizontal position errors to be constrained within about 500m indefinitely (see Section 5.6.2) [34]. Such systems have been deployed in high-level reconnaissance aircraft and long-range bombers.

Stellar navigation may also be used alongside accelerometer leveling (Sections 5.5.2 and 10.1.1) to determine position. With accelerometers accurate to $100 \mu\text{g}$, the latitude and longitude may be determined from C_e^n or C_e^w to within about 600m.

11.4.2 Gravity Gradiometry

The acceleration due to gravity varies slightly over a few kilometers due to variations in the terrain height because higher terrain contains more matter, from which gravitational attraction arises. It also varies over tens of kilometers as a result of geological formations, as different materials have different densities.

In principle, position may be determined by measuring the acceleration due to gravity and matching a succession of measurements with a high-resolution gravity database. However, in practice, better performance is obtained by matching the spatial gravity gradient, $\partial g/\partial r$. This is because the signal to noise of the local variations is greater in gradient measurements, which are also less susceptible to disturbance from vehicle motion. The spatial gravity gradient, $\partial g/\partial r$, has nine

components. However, only six of these are different as \mathbf{g} is itself the spatial derivative of the gravity potential, V_g , and the two spatial differentiation operations commute.

A gravity gradiometer measures each component of the gravity gradient by differencing the outputs of a pair of accelerometers with parallel sensitive axes. The host vehicle acceleration and the constant component of gravity cancel out. The instrument biases may be cancelled by mounting the accelerometers on a rotating disc and processing the outputs accordingly [36]. However, this does limit the bandwidth of the gravity gradient measurements. A typical rotation rate is 0.25 Hz, giving a sensor bandwidth of 0.06 Hz [37].

Gravity gradiometry is commonly deployed aboard military submarines alongside a high-performance INS and depth sensor, limiting the position error to about 400m [36]. It has the advantage over sonar of maintaining covertness.

In principle, gravity gradiometry could be used for aircraft positioning to an accuracy of tens of meters, similar to that achievable with TRN [38]. However, the resolution achievable with a submarine gradiometer on an aircraft is limited to about 2 km. This is partly due to bandwidth limitations and partly because much greater sensitivity is needed in the air, as the size of the gravity gradient diminishes as the cube of the distance from the terrain features producing that gradient [37].

11.4.3 Magnetic Field Variation

The Earth's magnetic field (see Section 10.1.2) varies with location, so if the host vehicle attitude, including heading, can be determined by other means, the position can be obtained by comparing the measured inclination and dip angles with a global magnetic model. For orbital spacecraft, the Earth's magnetic field is regular, enabling position to be determined within a few hundred meters [39].

For air and marine applications, local anomalies, which may be time varying, limit the position accuracy achievable from the global geomagnetic field to a few kilometers. However, by measuring the variations in the magnetic field over a few kilometers and matching this to a database, positions accurate to about a kilometer can be obtained for low-altitude aircraft [40].

References

- [1] Robins, A. J., "Recent Developments in the TERPROM Integrated Navigation System," *Proc. ION 44th AM*, Annapolis, MD, June 1988.
- [2] Golden, J. P., "Terrain Contour Matching (TERCOM): A Cruise Missile Guidance Aid," *Image Processing for Missile Guidance*, SPIE Vol. 238, 1980, pp. 10–18.
- [3] El-Sheimy, N., C. Valeo, and A. Habib, *Digital Terrain Modeling: Acquisition, Manipulation and Applications*, Norwood, MA: Artech House, 2005.
- [4] Perrett, M., and J. Krempasky, "Terrain Aiding for Precision Navigation in Heavy GPS Jamming," *Proc. ION GPS 2001*, Salt Lake City, UT, September 2001, pp. 924–931.
- [5] Boozer, D. D., and J. R. Fellerhoff, "Terrain-Aided Navigation Test Results in the AFTI/F-16 Aircraft," *Navigation: JION*, Vol. 35, No. 2., 1988, pp. 161–175.

- [6] Yu, P., Z. Chen, and J. C. Hung, "Performance Evaluation of Six Terrain Stochastic Linearization Techniques for TAN," *Proc. IEEE National Aerospace and Electronics Conference*, Dayton, OH, May 1991, pp. 382–388.
- [7] Enns, R., and D. Morrell, "Terrain-Aided Navigation Using the Viterbi Algorithm," *Journal of Guidance, Control and Dynamics*, Vol. 18, No. 6, 1995, pp. 1444–1449.
- [8] Bergman, N., L. Ljung, and F. Gustafsson, "Terrain Navigation Using Bayesian Statistics," *IEEE Control Systems Magazine*, June 1999, pp. 33–40.
- [9] Nordlund, P.-J., and F. Gustafsson, "Recursive Estimation of Three-Dimensional Aircraft Position Using Terrain Aided Positioning," *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Orlando, FL, 2002, pp. II-1121–1124.
- [10] Metzger, J., and J. F. Trommer, "Improvement of Modular Terrain Navigation Systems by Measurement Decorrelation," *Proc. ION 59th AM*, Albuquerque, NM, June 2003, pp. 353–362.
- [11] Hollowell, J., "HELI/SITAN: A Terrain Referenced Navigation Algorithm for Helicopters," *Proc. IEEE PLANS*, Las Vegas, NV, March 1990, pp. 616–625.
- [12] Runnalls, A. R., "A Bayesian Approach to Terrain Contour Navigation," *Proc. NATO AGARD 40th Guidance and Control Panel Symposium*, May 1985, paper 43.
- [13] Groves, P. D., R. J. Handley, and A. R. Runnalls, "Optimising the Integration of Terrain Referenced Navigation with INS and GPS," *Journal of Navigation*, Vol. 59, No. 1, 2006, pp. 71–89.
- [14] Runnalls, A. R., P. D. Groves, and R. J. Handley, "Terrain-Referenced Navigation Using the IGMAP Data Fusion Algorithm," *Proc. ION 61st Annual Meeting*, Boston, MA, June 2005, pp. 976–987.
- [15] Runnalls, A.R., and R. J. Handley, "The 'Gold Standard' Navigator," *Proc. Eurofusion98*, Great Malvern, U.K., November 1998, pp. 77–82.
- [16] Bar-Gill, A., P. Ben-Ezra, and I. Y. Bar-Itzack, "Improvement of Terrain-Aided Navigation in a Trajectory Optimization," *IEEE Trans. on Control Systems Technology*, Vol. 2, No. 4, 1994, pp. 336–342.
- [17] Neregård, F., et al., "Saab TERNAV, an Algorithm for Real Time Terrain Navigation and Results from Flight Trials Using a Laser Altimeter," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 1136–1145.
- [18] Handley, R. J., et al., "Future Terrain Referenced Navigation Techniques Exploiting Sensor Synergy," *Proc. GNSS 2003*, ENC, Graz, Austria, April 2003.
- [19] Campbell, J. L., M. Uijt de Haag, and F. van Graas, "Terrain Referenced Positioning Using Airborne Laser Scanner," *Navigation: JION*, Vol. 52, No. 4, 2005, pp. 189–197.
- [20] Campbell, J. L., M. Uijt de Haag, and F. van Graas, "Terrain Referenced Precision Approach Guidance Proof-of-Concept Flight Test Results," *Navigation: JION*, Vol. 54, No. 1, 2007, pp. 21–29.
- [21] Soehren, W., and W. Hawkinson, "A Prototype Personal Navigation System," *Proc. IEEE/ION PLANS*, San Diego, CA, April 2006, pp. 539–546.
- [22] Lucido, L., et al., "A Terrain Referenced Underwater Positioning Using Sonar Bathymetric Profiles and Multiscale Analysis," *Proc. MTS/IEEE Oceans '96*, Fort Lauderdale, FL, September 1996, pp. 417–421.
- [23] Strauss, O., F. Comby, and M. J. Aldon, "Multibeam Sonar Image Matching for Terrain-Based Underwater Navigation," *Proc. MTS/IEEE Oceans '99*, September 1999, pp. 882–887.
- [24] Bevington, J. E., and C. A. Marttila, "Precision Aided Inertial Navigation Using SAR and Digital Map Data," *Proc. IEEE PLANS*, Las Vegas, NV, March 1990, pp. 490–496.
- [25] Sim, D.-G. et al., "Hybrid Estimation of Navigation Parameters from Aerial Image Sequence," *IEEE Trans. on Image Processing*, Vol. 8, No. 3, 1999, pp. 429–435.
- [26] Handley, R. J., J. P. Abbott, and C. R. Surawy, "Continuous Visual Navigation—An Evolution of Scene Matching," *Proc. ION NTM*, Long Beach, CA, January 1998, pp. 217–224.

- [27] Handley, R. J., L. Dack, and P. McNeil, "Flight Trials of the Continuous Visual Navigation System," *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 185–192.
- [28] Zhao, Y., *Vehicle Location and Navigation Systems*, Norwood, MA: Artech House, 1997.
- [29] Yu, M., et al., "Improvement on Integrity and Reliability of Vehicle Positioning by a New Map Matching Method," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 2086–2094.
- [30] Pyo, J.-S., D.-H. Shin, and T.-Y. Sung, "Development of a Map Matching Method Using the Multiple Hypothesis Technique," *Proc. IEEE Intelligent Transportation Systems Conference*, Oakland, CA, August 2001, pp. 23–27.
- [31] Jagadeesh, G. R., T. Srikathan, and X. D. Zhang, "A Map Matching Method for GPS Based Real-Time Vehicle Location," *Journal of Navigation*, Vol. 57, No. 3, 2004, pp. 429–440.
- [32] Ochieng, W. Y., M. A. Quddus, and R. B. Noland, "Integrated Positioning Algorithms for Transport Telematics Applications," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 692–705.
- [33] Mueller, T., et al., "Design and Testing of a Robust High Speed Rail Prototype GPS Locomotive Location System," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 729–740.
- [34] Knobbe, E. J., and G. N. Haas, "Celestial Navigation," in *Avionics Navigation Systems*, 2nd ed., M. Kayton and W. R. Fried, (eds.), New York: Wiley, 1997, pp. 551–596.
- [35] Liebe, C. C., "Star Trackers for Attitude Determination," *IEEE AES Magazine*, June 1994, pp. 10–16.
- [36] Jircitano, A., and D. E. Dosch, "Gravity Aided Inertial Navigation System (GAINS)," *Proc. ION 47th AM*, June 1991, pp. 221–229.
- [37] Gleason, D. M., "Passive Airborne Navigation and Terrain Avoidance Using Gravity Gradiometry," *Journal of Guidance, Control and Dynamics*, Vol. 18, No. 6, 1995, pp. 1450–1458.
- [38] Affleck, C. A., and A. Jircitano, "Passive Gravity Gradiometer Navigation System," *Proc. IEEE PLANS*, Las Vegas, NV, March 1990, pp. 60–66.
- [39] Zuo, W., and F. Song, "An Autonomous Navigation Scheme Using Global Positioning System/Geomagnetism Integration for Small Satellites," *Proc. Institute of Mechanical Engineers Vol. 214G*, No. 4, 2000, pp. 207–215.
- [40] Wilson, J. M., et al., "Passive Navigation Using Local Magnetic Field Variations," *Proc. ION NTM*, Monterey, CA, January 2006, pp. 770–779.

Selected Bibliography

Taylor, G., and G. Blewitt, *Intelligent Positioning: GIS-GPS Unification*, New York: Wiley, 2006.

PART IV

Integrated Navigation

INS/GNSS Integration

Inertial navigation (Chapter 5) has a number of advantages. It operates continuously, bar hardware faults; provides high-bandwidth output at at least 50 Hz; and exhibits low short-term noise. It also provides effective attitude, angular rate, and acceleration measurements, as well as position and velocity. However, the accuracy of an inertial navigation solution degrades with time as the inertial instrument errors are integrated through the navigation equations, while INS capable of providing effective sole-means navigation for more than a few minutes after initial alignment are expensive at around \$100,000 or 100,000 Euros.

GNSS (Chapters 6 to 8) provides a high long-term position accuracy with errors limited to a few meters (stand-alone), while user equipment is available for less than \$/€ 100. However, compared to INS, the output rate is low, typically around 10 Hz, the short-term noise of a code-based position solution is high, and standard GNSS user equipment does not measure attitude. GNSS signals are also subject to obstruction and interference, so GNSS cannot be relied upon to provide a continuous navigation solution.

The benefits and drawbacks of INS and GNSS are complementary, so by integrating them, the advantages of both technologies are combined to give a continuous, high-bandwidth, complete navigation solution with high long- and short-term accuracy. In an integrated INS/GNSS, or GNSS/INS, navigation system, GNSS measurements prevent the inertial solution drifting, while the INS smoothes the GNSS solution and bridges signal outages.

INS/GNSS integration is suited to established inertial navigation applications such as ships, airliners, military aircraft, and long-range missiles. Integration with GNSS also makes inertial navigation practical with lower cost tactical-grade inertial sensors (see Chapter 4), making INS/GNSS a suitable navigation solution for light aircraft, helicopters, UAVs, short- and medium-range guided weapons, smaller boats, and potentially trains. INS/GNSS is sometimes used for road vehicles and personal navigation. However, lower cost dead-reckoning techniques, such as odometers and PDR, are often integrated with GNSS instead as discussed in Section 14.3.

Figure 12.1 shows the basic configuration of a typical INS/GNSS navigation system. The integration algorithm compares the inertial navigation solution with the outputs of GNSS user equipment and estimates corrections to the inertial position, velocity, and attitude solution, usually alongside other parameters. It is usually based on a Kalman filter, described in Chapter 3. The corrected inertial navigation solution then forms the integrated navigation solution. This architecture ensures that an integrated navigation solution is always produced, regardless of

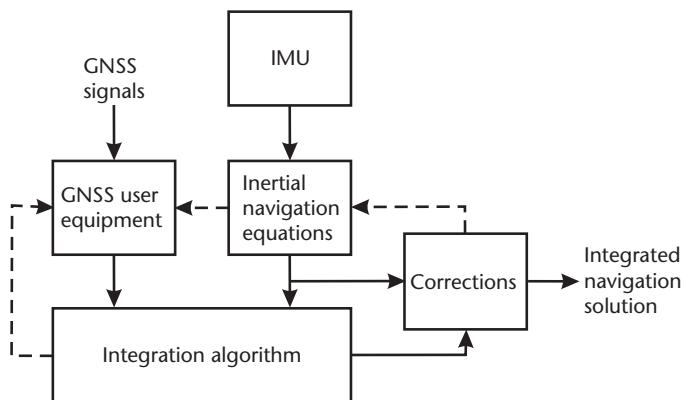


Figure 12.1 Generic INS/GNSS integration architecture.

GNSS signal availability. The dotted lines in Figure 12.1 show data flows present in some systems but not others; these are discussed later.

The hardware configuration of INS/GNSS systems varies. The integration algorithm may be hosted in the INS, the GNSS user equipment, or separately. Alternatively, everything may be hosted in one unit, sometimes known as an embedded GNSS in INS (EGI). Where the inertial navigation equations and integration algorithm share the same processor, but the IMU is separate, the system is sometimes known as an integrated IMU/GNSS or GNSS/IMU. However, an IMU/GNSS is no different to an INS/GNSS.

Section 12.1 describes and compares the different INS/GNSS integration architectures. Section 12.2 discusses state selection for INS/GNSS integration Kalman filters and describes typical system models, while Section 12.3 describes measurement models. Finally, Section 12.4 discusses advanced INS/GNSS implementations, such as those using differential and carrier-phase GNSS and GNSS attitude, handling large heading errors, performing smoothing, or using advanced inertial sensor modeling.

12.1 Integration Architectures

The architecture of an INS/GNSS integrated navigation system varies in three respects: how corrections are applied to the inertial navigation solution, what types of GNSS measurements are used, and how the GNSS user equipment is aided by the INS and integration algorithm. These are largely independent of each other. In the literature, terms such as loosely coupled, tightly coupled, ultratightly coupled, closely coupled, cascaded, and deep are used to define integration architectures [1–5]. However, there is no commonly agreed definition of these terms. Here, the most widely used definitions are adopted.

A *loosely coupled* INS/GNSS system uses the GNSS position and velocity solution as the measurement inputs to the integration algorithm, irrespective of the type of INS correction or GNSS aiding used. It is a cascaded architecture where the GNSS user equipment incorporates a navigation filter.

A *tightly coupled* INS/GNSS system uses the GNSS pseudo-range and pseudo-range-rate, delta-range, or ADR measurements as inputs to the integration algorithm, again irrespective of the type of INS correction or GNSS aiding used. The term closely coupled has been applied to both tightly and loosely coupled architectures, so is avoided here.

Deep integration, also known as ultratightly coupled (UTC), combines INS/GNSS integration and GNSS signal tracking into a single estimation algorithm. It uses the Is and Qs from the GNSS correlation channels as measurements, either directly or via discriminator functions, and generates the NCO commands used to control the reference code and carrier within the GNSS receiver (see Section 7.2.4).

The simplest way of combining INS and GNSS is an *uncoupled* system, whereby GNSS is simply used to reset the inertial navigation solution at intervals, often prompted by a manual command. This architecture has been applied in some aircraft where GPS was retrofitted when an INS was already installed. It is not true integration and is not discussed further.

The section begins with a description of the different methods of correcting the inertial navigation solution: open-loop correction, closed-loop correction, and total-state integration. The loosely coupled and tightly coupled architectures are then described and compared, followed by a discussion of GNSS aiding with these architectures. Finally, the deep integration architecture is described.

12.1.1 Correction of the Inertial Navigation Solution

The integrated navigation solution of an INS/GNSS integrated navigation system is the corrected inertial navigation solution. In a conventional integration architecture using an error-state Kalman filter and separate inertial navigation processing, correction may be either open-loop or closed-loop, regardless of what type of GNSS measurements are used and how the GNSS user equipment is aided. Both correction architectures are shown in Figure 12.2.

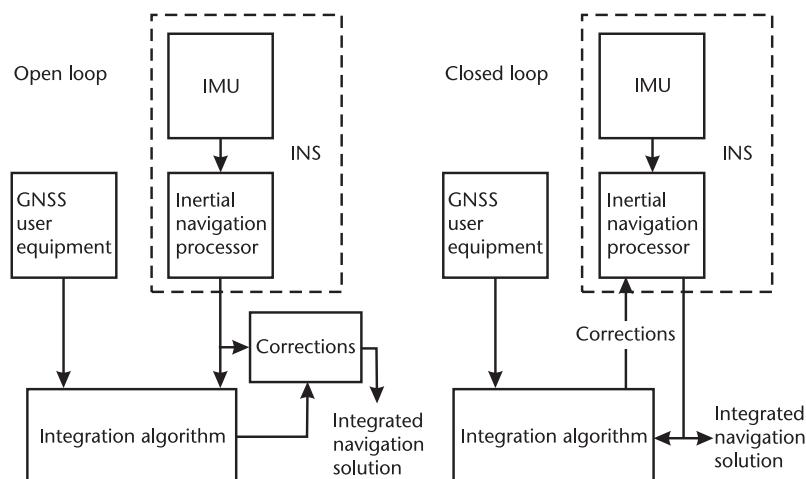


Figure 12.2 Open- and closed-loop INS correction architectures. (From: [6]. © 2002 QinetiQ Ltd. Reprinted with permission.)

In the open-loop configuration, the estimated position, velocity, and attitude errors are used to correct the inertial navigation solution within the integration algorithm at each iteration but are not fed back to the INS. Consequently, only the integrated navigation solution contains the Kalman filter estimates, and a raw INS solution is available for use in integrity monitoring³ (see Section 15.4.2). Either the raw INS or integrated navigation solution may be used for GNSS aiding.

The corrected inertial navigation solution, $\hat{\mathbf{C}}_b^\gamma$, $\hat{\mathbf{v}}_{\beta b}^\gamma$, and $\hat{\mathbf{r}}_{\beta b}^\gamma$ or $\hat{\mathbf{p}}_b$, which forms the integrated navigation solution, is obtained from the raw inertial navigation solution, $\tilde{\mathbf{C}}_b^\gamma$, $\tilde{\mathbf{v}}_{\beta b}^\gamma$, and $\tilde{\mathbf{r}}_{\beta b}^\gamma$ or $\tilde{\mathbf{p}}_b$, using

$$\hat{\mathbf{C}}_b^\gamma = \delta \hat{\mathbf{C}}_b^{\gamma T} \tilde{\mathbf{C}}_b^\gamma \quad (12.1)$$

$$\hat{\mathbf{v}}_{\beta b}^\gamma = \tilde{\mathbf{v}}_{\beta b}^\gamma - \delta \hat{\mathbf{v}}_{\beta b}^\gamma \quad (12.2)$$

and

$$\hat{\mathbf{r}}_{\beta b}^\gamma = \tilde{\mathbf{r}}_{\beta b}^\gamma - \delta \hat{\mathbf{r}}_{\beta b}^\gamma \quad (12.3)$$

or

$$\begin{aligned} \hat{L}_b &= \tilde{L}_b - \delta \hat{L}_b \\ \hat{\lambda}_b &= \tilde{\lambda}_b - \delta \hat{\lambda}_b \\ \hat{h}_b &= \tilde{h}_b - \delta \hat{h}_b \end{aligned} \quad (12.4)$$

where the attitude, velocity, and position errors, $\delta \hat{\mathbf{C}}_b^\gamma$, $\delta \hat{\mathbf{v}}_{\beta b}^\gamma$, $\delta \hat{\mathbf{r}}_{\beta b}^\gamma$, $\delta \hat{L}_b$, $\delta \hat{\lambda}_b$, and $\delta \hat{h}_b$, are as defined by (5.93) to (5.95), with $\hat{\cdot}$ denoting a Kalman filter estimate. The reference frame, β , and resolving axes, γ , are given by

$$\{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (12.5)$$

and depend on which coordinate frames are used for the inertial navigation equations (see Chapter 5).

Where the small angle approximation is applicable to the attitude errors, which is often not the case with open-loop integration, (12.1) becomes

$$\hat{\mathbf{C}}_b^\gamma \approx (\mathbf{I}_3 - [\delta \hat{\boldsymbol{\psi}}_{\gamma b}^\gamma \wedge]) \tilde{\mathbf{C}}_b^\gamma \quad (12.6)$$

where $\delta \hat{\boldsymbol{\psi}}_{\gamma b}^\gamma$ is the Kalman filter estimate of the attitude error of the INS body frame, b , with respect to frame γ , resolved about the frame γ axes.

In the closed-loop configuration, the estimated position, velocity, and attitude errors are fed back to the inertial navigation processor, either on each Kalman filter iteration or periodically, where they are used to correct the inertial navigation solution itself. The Kalman filter's position, velocity, and attitude estimates are zeroed after each set of corrections is fed back. Consequently, there is no independent uncorrected inertial navigation solution. As discussed in Section 3.2.6, a closed-

loop Kalman filter minimizes the size of the states, minimizing the linearization errors in the system model.

In closed-loop integration, there is only the corrected inertial navigation solution. New corrections are applied using

$$\hat{\mathbf{C}}_b^\gamma(+) = \delta\hat{\mathbf{C}}_b^{\gamma T}\hat{\mathbf{C}}_b^\gamma(-) \approx (\mathbf{I}_3 - [\delta\hat{\boldsymbol{\psi}}_{\gamma b}^\wedge])\hat{\mathbf{C}}_b^\gamma(-) \quad (12.7)$$

$$\hat{\mathbf{v}}_{\beta b}^\gamma(+) = \hat{\mathbf{v}}_{\beta b}^\gamma(-) - \delta\hat{\mathbf{v}}_{\beta b}^\gamma \quad (12.8)$$

and

$$\hat{\mathbf{r}}_{\beta b}^\gamma(+) = \hat{\mathbf{r}}_{\beta b}^\gamma(-) - \delta\hat{\mathbf{r}}_{\beta b}^\gamma \quad (12.9)$$

or

$$\begin{aligned} \hat{L}_b(+) &= \hat{L}_b(-) - \delta\hat{L}_b \\ \hat{\lambda}_b(+) &= \hat{\lambda}_b(-) - \delta\hat{\lambda}_b \\ \hat{h}_b(+) &= \hat{h}_b(-) - \delta\hat{h}_b \end{aligned} \quad (12.10)$$

where the suffixes (-) and (+) denote before and after the correction, respectively, and the small angle approximation is usually applicable to the attitude error.

In the closed-loop integration architecture, any accelerometer and gyro errors estimated by the Kalman filter are fed back to correct the IMU measurements, using (4.18), as they are input to the inertial navigation equations. These corrections are in addition to any that may be applied by the IMU's processor. Unlike the position, velocity, and attitude corrections, the accelerometer and gyro corrections must be applied on every iteration of the navigation equations, with feedback from the Kalman filter periodically updating the accelerometer and gyro errors. It may either feed back replacement estimates to the navigation equations, or estimate residual errors and feed back perturbations to the error estimates stored by the navigation equations. In the latter case, the Kalman filter estimates are zeroed on feedback.

The choice of open- or closed-loop INS/GNSS integration is a function of both the INS quality and the integration algorithm quality. Where low-grade inertial sensors are used, only the closed-loop configuration is suitable, regardless of the integration algorithm quality. This is because the raw inertial navigation solution will be of little use, while an open-loop configuration is likely to lead to large linearization errors in the Kalman filter. Conversely, where a high-quality INS is used with a low-quality integration algorithm, an open-loop configuration should be used as integrity monitoring is likely to be needed, whereas linearization errors will be small. Alternatively, a raw inertial navigation solution may be maintained in parallel to a closed-loop integrated solution. Where both the INS and the integration algorithm are high quality, the open and closed-loop configurations are both suitable.

Navigation systems where the IMU is supplied separately and the inertial navigation equations and algorithms share a common processor are ideally suited

to closed-loop operation, as the feedback of corrections is fully under the control of the integrated navigation system designer. However, where the INS is supplied as a complete unit, closed-loop integration should be approached with caution, as it is then necessary to ensure that the corrections are sent in the form that the INS is expecting, which may not be clearly defined.

The alternative to an error-state Kalman filter in INS/GNSS integration is a total-state Kalman filter, which estimates absolute position, velocity, and attitude instead of the errors in the corresponding INS outputs. In a total-state Kalman filter, the inertial navigation equations are embedded in the system model. As these are nonlinear, an extended Kalman filter (Section 3.4.1) must be used. The system model is then a function of the IMU outputs [7]. Figure 12.3 shows the system architecture.

In total-state integration, the system model must be iterated at the same rate as the inertial navigation equations in an error-state implementation. However, the processor load can be limited by iterating the system propagation of the error covariance matrix, P , at a lower rate than that of the state vector, x . The equations processed in a total-state INS/GNSS implementation are the same as those in a closed-loop error-state implementation, so the performance will be the same. The difference lies in the software architecture.

12.1.2 Loosely Coupled Integration

Figure 12.4 shows a loosely coupled INS/GNSS integration architecture, an example of cascaded integration (see Section 14.1.2). The position and/or velocity from the GNSS navigation solution (Section 7.5) is input as a measurement to the integration Kalman filter, which uses it to estimate the INS errors. The integrated navigation solution is the INS navigation solution, corrected with the Kalman filter estimates of its errors.¹

Generally, use of velocity rather than position measurements improves the observability of INS attitude and instrument errors. This is because these errors are fewer integration/differentiation steps away from velocity than position in the system and measurement models. Thus, using velocity measurements reduces the lag in estimating these states, though there is no additional information. However, use of velocity measurements reduces the observability of the position error as measurement noise is integrated up into the state estimates. Therefore, most INS/GNSS integration algorithms use both velocity and position measurements.

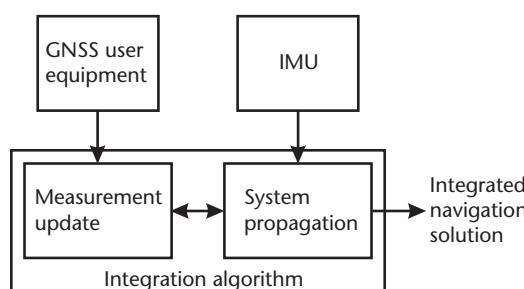


Figure 12.3 Total-state INS/GNSS integration architecture.

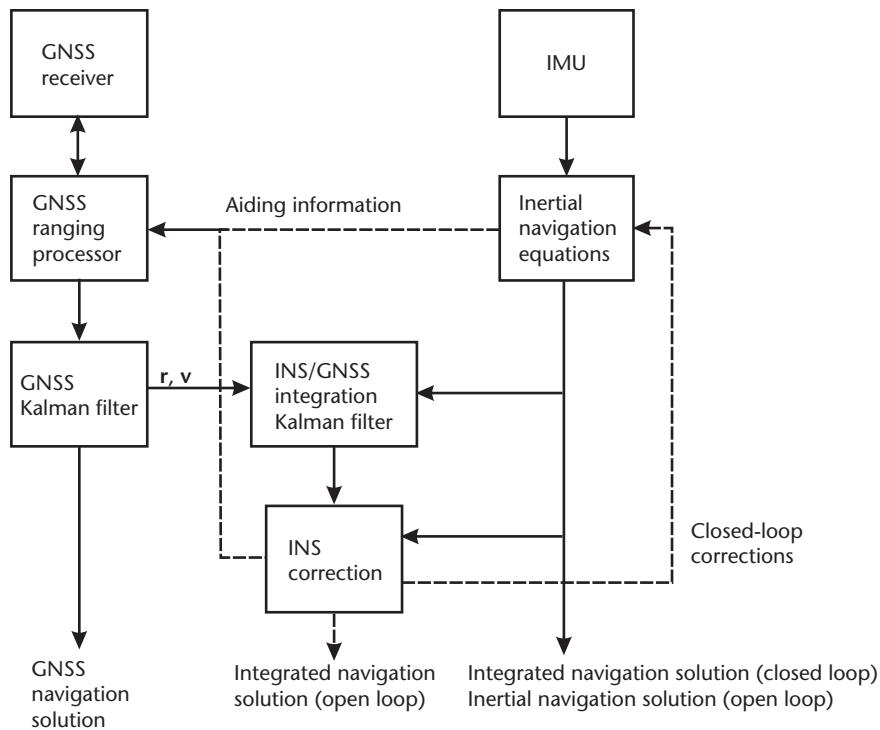


Figure 12.4 Loosely coupled INS/GNSS integration architecture.

The two main advantages of loosely coupled integration are simplicity and redundancy. The architecture is simple in that it can be used with any INS and any GNSS user equipment, making it particularly suited to retrofit applications. In a loosely coupled architecture, there is usually a stand-alone GNSS navigation solution available in addition to the integrated solution.² Where open-loop INS correction is implemented, there is also an independent INS solution. This enables basic parallel solutions integrity monitoring (see Section 15.4.2).

 The main problem with loosely coupled INS/GNSS integration stems from the use of cascaded Kalman filters (i.e., the fact that the output of a GNSS Kalman filter may be used as a measurement input to the integration Kalman filter. The errors of Kalman filter outputs are time correlated, whereas Kalman filter measurement errors are assumed to be uncorrelated in time. As discussed in Section 3.4.2, input of time-correlated measurements can disrupt Kalman filter state estimation unless the filter gain is reduced or the correlated errors are estimated. The correlation time of the GNSS navigation-solution errors varies and can be up to 100 seconds on the position and 20 seconds on the velocity. This is too short for the correlated errors to be estimated, but long enough to significantly slow down the estimation of the INS errors.¹

Selection of the integration Kalman filter gain and measurement iteration rate is critical. If measurements are processed too quickly, the filter is liable to become unstable. Conversely, if measurements are processed too slowly, the observability of the INS errors will be reduced.² For stability, the system must be tuned so that



the integration Kalman filter bandwidth is always less than that of the GNSS Kalman filter, noting that the bandwidths vary. Measurement-update intervals of 10s are common in loosely coupled systems.

This problem does not arise when the GNSS user equipment computes a single-point navigation solution, provided the update interval exceeds the tracking-loop time constants. However, single-point navigation solutions are much noisier, so a low gain is still needed for the integration Kalman filter.

There are further problems with the loosely coupled approach. Signals from four different satellites are required to maintain a GNSS navigation solution, though a solution can be maintained for short periods with three satellites. Consequently, where fewer satellites are tracked, the GNSS data cannot be used to aid the INS. Also, the integration filter needs to know the covariance of the GNSS filter output, as this varies with satellite geometry and availability. Many designs of GNSS user equipment do not output realistic covariances and some output no covariance information at all.³

12.1.3 Tightly Coupled Integration

Figure 12.5 shows a tightly coupled INS/GNSS integration architecture, an example of centralized integration (see Section 14.1.3). Here, the GNSS Kalman filter is subsumed into the INS/GNSS integration filter. The pseudo-range and pseudo-range rates from the GNSS ranging processor (Section 7.3) are input as measurements to the Kalman filter, which uses them to estimate the errors in the INS and GNSS

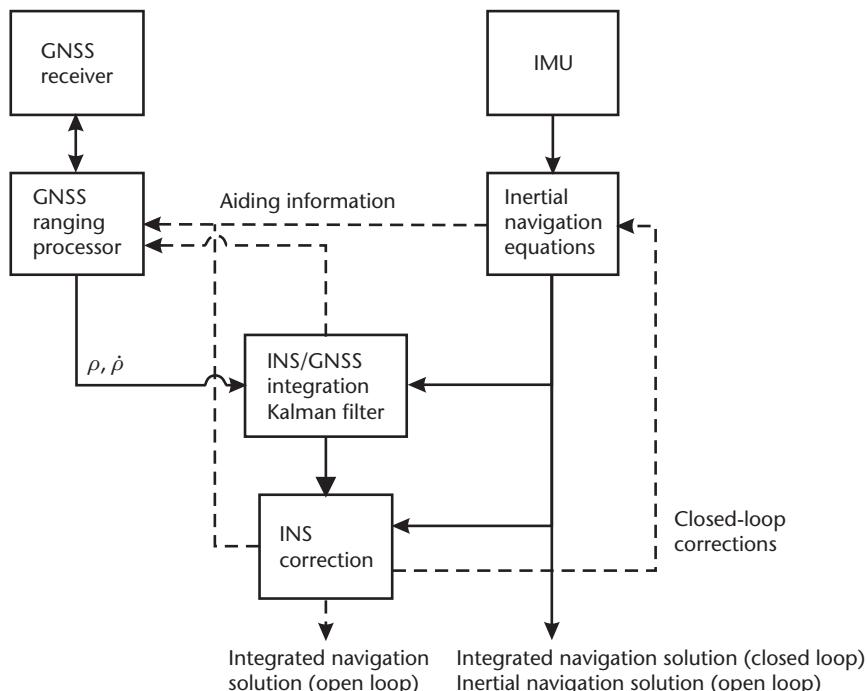


Figure 12.5 Tightly coupled INS/GNSS integration architecture.

systems. As with a loosely coupled architecture, the corrected inertial navigation solution forms the integrated navigation solution.¹

In theory, either pseudo-range or pseudo-range-rate measurements may be used, but in all practical systems, both are used because of the observability benefits described in Section 12.1.2 and because they are complementary: pseudo-ranges come from code tracking, whereas pseudo-range rates are derived from the more accurate, but less robust, carrier tracking.

The benefits of the tightly coupled architecture derive from combining the two Kalman filters of the loosely coupled architecture into one. The statistical problems of using the solution from one Kalman filter as the measurement of another are eliminated.² However, the Kalman filter bandwidths must still be kept within the GNSS tracking-loop bandwidths to prevent time-correlated tracking noise from contaminating the state estimates.

Furthermore, the handover of GNSS position and velocity covariance, due to satellite geometry and availability, to the integration algorithm is done implicitly. Finally, the system does not need a full GNSS solution to aid the INS. GNSS measurement data is still input even if only one satellite signal is tracked.¹

The main disadvantage of the tightly coupled architecture is that there is no inherent stand-alone GNSS solution. However, a GNSS-only navigation solution may be maintained in parallel where required. Given the same inertial instruments and the same GNSS user equipment, a tightly coupled INS/GNSS almost always performs better than its loosely coupled counterpart in terms of both accuracy and robustness.²

12.1.4 GNSS Aiding

In loosely and tightly coupled integration, the inertial navigation solution may be used to aid GNSS acquisition and tracking [8]. In deep integration, aiding of GNSS acquisition is the same, but tracking aiding is an inherent part of the integration architecture. Where open-loop INS correction is used, either the raw or corrected inertial navigation solution may be used for GNSS acquisition and tracking aiding.

The corrected solution is generally more accurate, but the raw solution is wholly independent of GNSS (after initialization) so is not subject to positive-feedback-induced errors. In tightly coupled and deep integration, the receiver clock may also be corrected.

Acquisition aiding provides the GNSS ranging processor with the approximate position and velocity, limiting the number of cells that need to be searched to acquire the signal (see Section 7.3.1). For reacquisition, where the satellite position and velocities are known and the receiver clock is calibrated, the number of cells to search can be very small, allowing very long dwell times. Simulations have shown that inertially aided reacquisition is feasible at C/N_0 levels down to about 10 dB-Hz [9].

GNSS tracking-loop bandwidths are a tradeoff between dynamics response and noise resistance (see Sections 7.3.2, 7.3.3, and 7.4.3). However, if the tracking loops are aided with the inertial navigation solution, they only have to track the receiver clock noise and the error in the INS solution, not the absolute dynamics of the user antenna. This enables narrower tracking-loop bandwidths to be used,

improving noise resistance and allowing tracking to be maintained at a lower C/N_0 [10].

However, the downside of narrow tracking bandwidths is longer error correlation times on the GNSS measurements input to the integration algorithm. This requires lower gains to be used in the Kalman filter to prevent instability (see Section 3.4.2). With an aviation-grade INS, this does not present a problem. However, with tactical-grade inertial sensors, a reduced-gain Kalman filter leads to poorer inertial calibration and larger navigation-solution errors. One solution is adaptive tightly coupled (ATC) integration [11, 12], where the tracking-loop bandwidths are varied according to the measured c/n_0 and the measurement noise covariance in the integration algorithm adapted to the tracking bandwidths so that the Kalman filter gains are matched to the tracking-error correlation times. ATC enables GNSS code to be tracked at a C/N_0 around 8 dB-Hz lower than a conventional tightly coupled system tuned for optimum INS calibration [11].

Inertial aiding of the code-tracking loop is normally implemented as a reversionary mode to carrier aiding. The pseudo-range rate estimated from the inertial navigation solution is

$$\hat{\dot{\rho}}_{Rj} = \hat{\mathbf{u}}_{as,j}^i{}^T (\hat{\mathbf{v}}_{is,j}^i(t_{st,j}) - \hat{\mathbf{v}}_{ia}^i(t_{sa})) + \delta\hat{\dot{\rho}}_{rc} - \Delta\dot{\rho}_{scj} \quad (12.11)$$

where the notation is as defined in Section 7.1.2. The receiver clock drift estimate, $\delta\hat{\dot{\rho}}_{rc}$, is obtained from the integration Kalman filter in tightly coupled integration and the GNSS Kalman filter in loosely coupled integration. The satellite velocity, $\hat{\mathbf{v}}_{is,j}^i$, is obtained from the navigation data message (see Section 7.1.1). The line-of-sight vector, $\hat{\mathbf{u}}_{as,j}^i$, is given by (7.34) and the satellite clock drift correction, $\Delta\dot{\rho}_{scj}$, obtained by differentiating (7.131). The GNSS antenna velocity is, from (2.109),

$$\hat{\mathbf{v}}_{ia}^i = \hat{\mathbf{v}}_{ib}^i + \hat{\mathbf{C}}_b^i (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) \quad (12.12)$$

where \mathbf{l}_{ba}^b is the lever arm from the IMU to the antenna, resolved about the IMU body frame. See Section 7.1.2 for ECEF-frame equivalents of (12.11). When carrier tracking is lost, the inertial aiding information must also be used to control the carrier NCO in order to maintain signal coherence over the correlator accumulation interval in the receiver and GNSS ranging processor (see Section 7.2.4.4). For a 20-ms accumulation interval, the pseudo-range rate must be accurate to within about 4 m s^{-1} .

Inertial aiding can also be used to maintain synchronization of the reference code phase and carrier frequency through short signal blockages, enabling tracking to resume when the signal returns without having to undergo reacquisition first. The key is to compensate for any loss of synchronization between loss of signal and detection of the loss of tracking lock. All-channel outages may be bridged for several tens of seconds and single-channel outages for several minutes [9].

Inertial and carrier aiding of the code tracking loop may also be implemented in parallel, in which case, a weighted average of the two pseudo-range rates should be constructed, based on the respective error standard deviations. Inertial aiding

of carrier tracking is less common, largely because it requires tight time synchronization for high-dynamics applications [13]. One solution is to store and retrieve the precorrelation signal samples so that inertial aiding derived from contemporaneous IMU measurements may be used; however, this requires a software receiver.

The integration algorithm's receiver-clock estimates may be fed back to correct the receiver clock itself in analogy with closed-loop INS correction. The Kalman filter states are zeroed when feedback takes place, which can occur at every iteration, at regular intervals, or when the estimates exceed a predetermined threshold. Caution must be exercised in correcting for the effects of any lags in applying the clock corrections and disabling any clock feedback from the GNSS Kalman filter. As no approximations are made in implementing the clock states in the Kalman filter, closed-loop receiver clock correction has no impact on performance.

The corrected inertial navigation solution can also be used to aid GNSS integrity monitoring (Chapter 15) and detection of GNSS cycle slips (see Section 8.2.1). Cycle slips may be detected by comparing the changes in ADR with the INS estimated pseudo-range rate, provided the carrier tracking loops are not inertially aided [14].

12.1.5 Deep Integration

Deep INS/GNSS integration is the INS/GNSS equivalent of combined GNSS navigation and tracking, discussed in Section 7.5.3. Figure 12.6 shows the integration architecture with closed-loop INS correction. The code and carrier NCO commands are generated using the corrected inertial navigation solution, the satellite position and velocity from the navigation data message, and various GNSS error estimates. The accumulated correlator outputs from the GNSS receiver, the Is and Qs, are

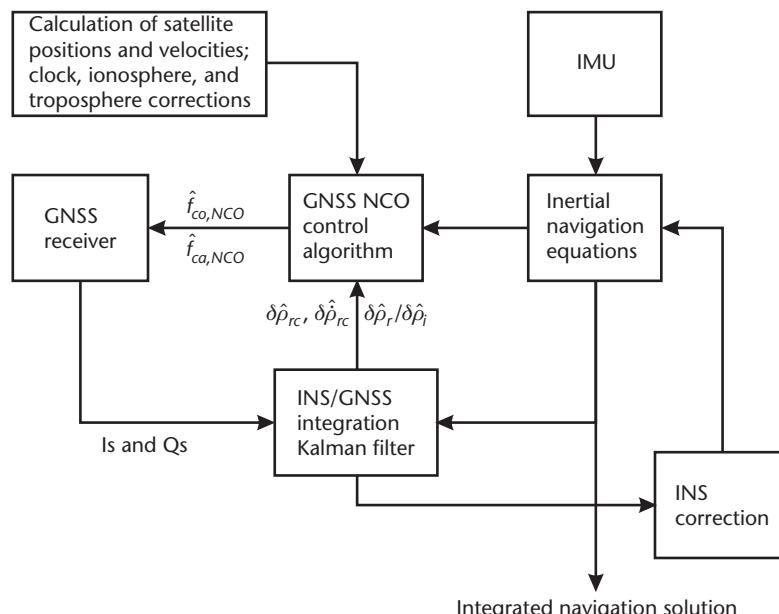


Figure 12.6 Deep INS/GNSS integration architecture (closed-loop INS correction).

input directly to the integration algorithm, usually Kalman filter based, where a number of INS and GNSS errors are estimated. The corrected inertial navigation solution forms the integrated navigation solution, as in the other architectures.

Compared with combined GNSS navigation and tracking, deep INS/GNSS integration has the advantage that only the errors in the INS solution need be tracked, as opposed to the absolute dynamics. This enables a lower tracking bandwidth to be used, increasing noise resistance. Deep integration can also operate with fewer than four GNSS satellites for limited periods.

Compared with tightly coupled integration, deep avoids the weighting down of the older I and Q measurement data when a pseudo-range or pseudo-range rate output interval is greater than the tracking-loop time constant and avoids the need to reduce the Kalman filter gain when it is not. This enables deep integration to operate at lower C/N_0 levels. Like ATC, a deep integration algorithm can adapt to different C/N_0 levels by varying the measurement weighting. Bridging code and carrier-frequency tracking through brief signal outages is inherent in deep integration. By removing the cascade between the tracking-loop filters and integration filter, deep offers an optimal integration architecture [15].

The Is and Qs must be output by the GNSS receiver at at least the navigation message symbol rate (50 Hz for the legacy GPS and GLONASS signals), while the NCO commands are usually input at the same rate. A faster data rate reduces communication lags but requires the integration algorithm to know where the navigation data-bit transitions are so that it can perform correct coherent summation of the Is and Qs. The need to implement a new and much faster interface between the GNSS user equipment and integration algorithm is the main drawback of deep integration and makes it difficult to retrofit.

There are two classes of deep integration algorithm: coherent and noncoherent [16]. Coherent deep integration inputs the Is and Qs as direct measurements to the Kalman filter, while noncoherent integration uses discriminator functions. Coherent integration is more accurate, as it avoids discriminator nonlinearities and reduces code-tracking noise to that obtained with a coherent discriminator (see Section 7.4.3). However, it can only operate where there is sufficient signal to noise to track carrier phase, so noncoherent deep integration is more robust.

Coherent deep integration requires the Kalman filter measurement update to be iterated at the navigation-data-message rate, while noncoherent deep integration does not. Thus, coherent deep integration can impose a much higher processing load. This tends to be reduced in practice by partitioning the Kalman filter, as described in Section 12.3.3.

The code and carrier NCO commands are given by (7.211) and (7.213), assuming the signal and reference carrier phases are not synchronized. The pseudo-range estimated from the inertial navigation solution is

$$\hat{\rho}_{Rj} = \left| \hat{r}_{is,j}(t_{st,j}) - \hat{r}_{ia}(t_{sa}) \right| + \delta\hat{\rho}_{rc} - \Delta\rho_{icj} - \Delta\rho_{tcj} - \Delta\rho_{scj} + \delta\hat{\rho}_{rj} \quad (12.13)$$

where the notation is as defined in Section 7.1.2, except for $\delta\hat{\rho}_{rj}$, which is the Kalman-filter-estimated residual range bias (see Section 12.2.1). The receiver clock offset estimate, $\delta\hat{\rho}_{rc}$, is obtained from the Kalman filter. The satellite position, $\hat{r}_{is,j}$, is obtained from the navigation data message as described in Section 7.1.1.

The satellite clock correction, $\Delta\rho_{scj}$, is given by (7.131); the troposphere correction, $\Delta\rho_{tcj}$, is obtained from a model; and the ionosphere correction, $\Delta\rho_{icj}$, is obtained from a model and/or the Kalman filter. The GNSS antenna position is, from (2.106),

$$\hat{\mathbf{r}}_{ia}^i = \hat{\mathbf{r}}_{ib}^i + \hat{\mathbf{C}}_b^i \mathbf{l}_{ba}^b \quad (12.14)$$

See Section 7.1.2 for ECEF-frame equivalents of (12.13). The pseudo-range rate is given by (12.11).

Where there is a significant lag between the time of validity of the inertial navigation solution used to generate the NCO commands and the application of those commands in the GNSS receiver, they may be predicted forward in time using an estimate of the pseudo-range acceleration. Control lags may be eliminated by using a software receiver.

Tests have shown that noncoherent deep INS/GPS integration can track code at C/N_0 levels of 8 dB-Hz or less [17, 18], a margin of at least 12 dB-Hz over conventional tightly coupled INS/GPS with wide tracking bandwidths. Carrier-phase tracking down to 15 dB-Hz C/N_0 has been demonstrated with coherent deep integration and navigation data-bit estimation [19].

12.2 System Model and State Selection

The system model of a Kalman filter is described in Section 3.2.4. For INS/GNSS integration, it depends on which quantities are estimated as Kalman filter states, which, in turn, depends on the application, inertial sensors, integration architecture, and choice of coordinate frame. State selection is thus described first. A typical INS state-propagation model is then derived in the ECI frame, followed by descriptions of the ECEF and local-navigation-frame equivalents and then the INS system noise. Discussion of GNSS state propagation and system noise completes the section.

There is no interaction between the INS and GNSS states in the system model; they only interact through the measurement model. Therefore, the system, transition, and system noise covariance matrices may be partitioned:

$$\mathbf{F} = \begin{pmatrix} \mathbf{F}_{INS} & 0 \\ 0 & \mathbf{F}_{GNSS} \end{pmatrix}, \quad \boldsymbol{\Phi} = \begin{pmatrix} \boldsymbol{\Phi}_{INS} & 0 \\ 0 & \boldsymbol{\Phi}_{GNSS} \end{pmatrix}, \quad \mathbf{Q} = \begin{pmatrix} \mathbf{Q}_{INS} & 0 \\ 0 & \mathbf{Q}_{GNSS} \end{pmatrix} \quad (12.15)$$

where

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_{INS} \\ \mathbf{x}_{GNSS} \end{pmatrix} \quad (12.16)$$

An error-state implementation is assumed here; for total-state INS/GNSS integration, the propagation of the position, velocity, and attitude errors should be replaced by the inertial navigation equations described in Chapter 5.

12.2.1 State Selection and Observability

All error-state INS/GNSS integration algorithms estimate the position and velocity errors. These may be expressed in the ECI frame as δr_{ib}^i and δv_{ib}^i , in the ECEF frame as δr_{eb}^e and δv_{eb}^e , or in the local navigation frame as δL_b , $\delta \lambda_b$, δh_b , and δv_{eb}^n . The coordinate frame used for the integration algorithm should match that used for the inertial navigation equations (Chapter 5).

For all but the highest grades of INS, there is significant benefit in estimating the attitude error which, here, is expressed as a small angle, resolved about the coordinate frame used for navigation, $\delta \psi_{yb}^\gamma$. It may also be resolved about the INS body frame axes, giving $\delta \psi_{yb}^b$, or expressed as a quaternion [20] or rotation vector [2]. The small angle approximation is only suited to attitude errors where closed-loop INS correction is applied or the INS is of a high grade. For example, applying this approximation to a 2° attitude error perturbs the state estimate by about 5 percent. The handling of large attitude errors is discussed in Section 12.4.3.

Except where they are very small, the accelerometer biases, \mathbf{b}_a , should always be estimated where the attitude errors are estimated. Conversely, the attitude errors should always be estimated where the accelerometer biases are estimated. Otherwise, the attitude error estimates are contaminated by the effects of the acceleration errors, or vice versa. This is because, in INS/GNSS integration, the attitude and acceleration errors are observed through the growth in the velocity and position error they produce (see Section 12.3.4). As described in Section 5.6, both types of errors lead to an initial linear growth in velocity error and quadratic growth in position error.

As (5.100) shows, the heading error only produces a velocity error when there is acceleration in the horizontal plane. Therefore, the navigation system's host must undergo significant maneuvering for the INS heading error to be observed and calibrated using GNSS measurements. When the navigation system is level and not accelerating, the vertical accelerometer bias is the only cause of growth in the vertical velocity error. This makes it the most observable of the accelerometer biases and, as a consequence, an INS/GNSS navigation solution exhibits lower vertical drift than horizontal drift during GNSS outages. The roll and pitch attitude errors and horizontal accelerometer biases are observed as linear combinations under conditions of constant acceleration and attitude. To fully separate the estimates of these states, the host-vehicle must turn, while a forward acceleration will separate the pitch error and forward accelerometer bias. The Kalman filter keeps a record of the correlations between its state estimates in the off-diagonal elements of its error covariance matrix, \mathbf{P} . This is used to separate the attitude and acceleration error estimates when their impact on the velocity error changes. Observability of INS error states is discussed more formally in [21]. Note that it is independent of the axes about which the states are resolved.

The choice of further inertial instrument errors to estimate depends on their effect on the position, velocity, and attitude solution. If an IMU error has a significant impact on the navigation accuracy, it will be observable. Conversely, if its impact is much less than that of the random noise (Section 4.4.3), which cannot be calibrated, it will not be observable. This depends on the IMU design and user dynamics. The next most observable errors are the gyro biases, \mathbf{b}_g , which produce

a quadratic growth in the velocity error with time. These are estimated in most INS/GNSS integration algorithms. Again, the heading component is more difficult to observe than the roll and pitch components.

Whether there is any benefit in estimating the accelerometer and gyro scale factor and cross-coupling errors (see Section 4.4.2) or the gyro g-dependent biases (Section 4.4.4) depends on the accelerations and angular rates exhibited by the host. For most air, land, sea, and space applications, these errors are not observable. Exceptions are highly dynamic applications, such as motor sports and some guided weapons. Gyro scale factor and cross-coupling errors can also be significant for aircraft performing circling movements and roll-stabilized guided weapons (see Section 5.6.3). Often, individual dynamics-dependent IMU errors are observable, but not the full set.

Sometimes, where the observability of scale factor and cross-coupling errors is borderline, their inclusion in the Kalman filter state vector can improve the estimation of the accelerometer and gyro biases through the use of a more representative model of the IMU in the Kalman filter. Separating the biases into separate static and dynamic states (see Section 4.4.1) can have a similar effect [11]. Alternatively, these errors may be modeled as correlated system noise using a Schmidt-Kalman filter as described in Section 3.4.2.

The choice of GNSS states to estimate depends on the integration architecture. In loosely coupled integration, no GNSS states are normally estimated. In tightly coupled and deep integration, the receiver clock offset, $\delta\rho_{rc}$, and drift, $\dot{\delta\rho}_{rc}$, must be estimated as described in Section 7.5.2 for a stand-alone GNSS navigation filter. In an integrated INS/GNSS, the specific force on the receiver's reference oscillator is known, so the clock g-dependent errors may be estimated by the Kalman filter where they have an impact on system performance.

In dual-frequency GNSS user equipment, the ionosphere propagation error is normally corrected by combining pseudo-range measurements made on different frequencies, as described in Section 7.4.2. Another way of performing the smoothing is to estimate the ionosphere propagation delays as Kalman filter states, inputting the pseudo-range measurements on different frequencies separately. In a tightly coupled architecture, this significantly increases the processor load without bringing performance benefits. However, in deep integration, it is the only way of performing dual-frequency ionosphere calibration. Either the total ionosphere propagation delays or the errors in ionosphere model predictions may be estimated.

The Kalman filter may also estimate the correlated range errors due to the ephemeris and the residual satellite clock, troposphere, and ionosphere errors (see Section 7.4). These range biases are partially observable where signals from more than four GNSS satellites are tracked. Their inclusion as states or correlated system noise leads to a more representative error covariance matrix, P . In deep integration, range bias estimation can be used to keep the reference codes aligned with their respective signal correlation peaks where the range biases are a significant proportion of the code chip length. If a relatively short correlation time is modeled, range bias states may also be used to absorb multipath errors, lessening their impact on the navigation solution. Where range biases are not estimated, the position and clock offset uncertainties should be modified as described in Section 7.5.2.4.

For coherent deep integration, the reference–signal carrier phase offsets, $\delta\phi_{ca,j}$ (given by (7.67)), for each channel tracked must be estimated as states. The carrier frequency tracking errors, $\delta f_{ca,j}$, their derivatives, $\delta \dot{f}_{ca,j}$, and the signal amplitude may also be estimated [22, 23].

12.2.2 INS State Propagation in the Inertial Frame

A state propagation model is developed here for a Kalman filter estimating attitude, velocity, and position errors referenced to and resolved in the ECI frame, together with the accelerometer and gyro biases. The INS partition of the state vector comprises the following 15 states

$$\mathbf{x}_{INS}^i = \begin{pmatrix} \delta\psi_{ib}^i \\ \delta\mathbf{v}_{ib}^i \\ \delta\mathbf{r}_{ib}^i \\ \mathbf{b}_a \\ \mathbf{b}_g \end{pmatrix} \quad (12.17)$$

where the superscript i is used to denote the ECI-frame implementation of the integration Kalman filter, consistent with the notation used in Section 7.5.2.

To obtain the INS system model, the time derivative of each Kalman filter state must be calculated. The attitude error propagation is derived first, followed by the velocity and position error propagation. The complete system model and the transition matrix are then presented.

In a real navigation system, the true values of the attitude, velocity, and other kinematic parameters are unknown. The best estimates available are the corrected INS-indicated parameters. Thus, for a generic parameter, y , the approximation $y \approx E(y) = \hat{y}$ must be made in deriving the system model. This also applies to the measurement model. Note that where closed-loop correction of INS errors is implemented, the raw and corrected INS outputs are the same, so for a generic parameter, y , $\tilde{y} = \hat{y}$.

12.2.2.1 Attitude Error Propagation

From (5.97), the time derivative of the small-angle attitude error, $\delta\psi_{ib}^i$, may be obtained by differentiating its coordinate transformation matrix counterpart:

$$[\delta\dot{\psi}_{ib}^i \wedge] \approx \delta\dot{\mathbf{C}}_{ib}^i \quad (12.18)$$

From (5.97),

$$\delta\dot{\mathbf{C}}_b^i = \tilde{\mathbf{C}}_b^i \mathbf{C}_i^b + \tilde{\mathbf{C}}_b^i \dot{\mathbf{C}}_i^b \quad (12.19)$$

while from (2.28)

$$\dot{\mathbf{C}}_b^i = \mathbf{C}_b^i \boldsymbol{\Omega}_{ib}^b \quad (12.20)$$

noting that $\boldsymbol{\Omega}_{ib}^b$ is the skew-symmetric matrix of the angular rate, $\boldsymbol{\omega}_{ib}^b$. Substituting (12.20) and (12.19) into (12.18) gives

$$[\delta\dot{\psi}_{ib}^i \wedge] \approx \tilde{\mathbf{C}}_b^i \tilde{\boldsymbol{\Omega}}_{ib}^b \mathbf{C}_b^i + \tilde{\mathbf{C}}_b^i \mathbf{C}_b^i \boldsymbol{\Omega}_{bi}^i \quad (12.21)$$

Applying (2.27) and rearranging

$$\tilde{\mathbf{C}}_b^i [\delta\dot{\psi}_{ib}^i \wedge] \mathbf{C}_b^i \approx \tilde{\boldsymbol{\Omega}}_{ib}^b - \boldsymbol{\Omega}_{ib}^b \quad (12.22)$$

A fundamental assumption of Kalman filters is that the system model is a linear function of the state estimates. For this to apply in practice, the product of any two Kalman filter states in the system model derivation must be negligible. Thus, applying the approximation $\delta\dot{\psi}_{ib}^i \delta\dot{\psi}_{ib}^i \approx 0$,¹

$$\tilde{\mathbf{C}}_b^i [\delta\dot{\psi}_{ib}^i \wedge] \mathbf{C}_b^i \approx [(\hat{\mathbf{C}}_b^i \delta\dot{\psi}_{ib}^i) \wedge] \quad (12.23)$$

Substituting (12.23) into (12.22), taking components of the skew-symmetric matrices and rearranging gives²

$$\delta\dot{\psi}_{ib}^i \approx \hat{\mathbf{C}}_b^i (\tilde{\boldsymbol{\omega}}_{ib}^b - \boldsymbol{\omega}_{ib}^b) = \hat{\mathbf{C}}_b^i \delta\boldsymbol{\omega}_{ib}^b \quad (12.24)$$

The error in the gyro triad's angular-rate output, $\delta\boldsymbol{\omega}_{ib}^b$, is given by (4.16) and (4.17). Where the biases are the only gyro errors modeled as Kalman filter states, (12.24) becomes

$$\delta\dot{\psi}_{ib}^i \approx \hat{\mathbf{C}}_b^i \mathbf{b}_g \quad (12.25)$$

12.2.2.2 Velocity Error Propagation

From (5.11) and (5.12), the time derivative of the ECI-frame velocity is

$$\dot{\mathbf{v}}_{ib}^i = \mathbf{f}_{ib}^i + \boldsymbol{\gamma}_{ib}^i \quad (12.26)$$

Thus, the time derivative of the velocity error is

$$\delta\dot{\mathbf{v}}_{ib}^i = \tilde{\mathbf{f}}_{ib}^i - \mathbf{f}_{ib}^i + \tilde{\boldsymbol{\gamma}}_{ib}^i - \boldsymbol{\gamma}_{ib}^i \quad (12.27)$$

The accelerometers measure specific force in body axes, so the error in $\tilde{\mathbf{f}}_{ib}^i$ is due to a mixture of accelerometer and attitude errors:

$$\tilde{\mathbf{f}}_{ib}^i - \mathbf{f}_{ib}^i = \tilde{\mathbf{C}}_b^i \tilde{\mathbf{f}}_{ib}^b - \mathbf{C}_b^i \mathbf{f}_{ib}^b \quad (12.28)$$

Assuming the products of Kalman filter states may be neglected,

$$\tilde{\mathbf{f}}_{ib}^i - \mathbf{f}_{ib}^i \approx \hat{\mathbf{C}}_b^i (\tilde{\mathbf{f}}_{ib}^b - \mathbf{f}_{ib}^b) + (\tilde{\mathbf{C}}_b^i - \mathbf{C}_b^i) \hat{\mathbf{f}}_{ib}^b \quad (12.29)$$

The error in the accelerometer triad's specific-force output, $\delta\mathbf{f}_{ib}^b$, is given by (4.15) and (4.17). Where the biases are the only accelerometer errors modeled as Kalman filter states,

$$\tilde{\mathbf{f}}_{ib}^b - \mathbf{f}_{ib}^b = \delta\mathbf{f}_{ib}^b \approx \mathbf{b}_{as} \quad (12.30)$$

Applying the small angle approximation to the attitude error, (5.95) and (5.97) give

$$\tilde{\mathbf{C}}_b^i - \mathbf{C}_b^i = (\delta\mathbf{C}_b^i - \mathbf{I}_3) \mathbf{C}_b^i \approx [\delta\boldsymbol{\psi}_{ib}^i \wedge] \mathbf{C}_b^i \quad (12.31)$$

Turning to the gravitational term in (12.27), from Section 2.3.5, this scales with height roughly as [(2.89) repeated]:

$$\gamma_{ib}^i \approx \frac{(r_{eS}^e(L_b))^2}{(r_{eS}^e(L_b) + h_b)^2} \gamma_0^i(L_b)$$

The variation of gravitation with latitude is small, so the effect of the latitude error on the gravitation model can be neglected. Also, the effects of errors in the centripetal gravity term are negligible for height errors of a few kilometers, so the gravitation model may be approximated by a gravity model for most applications. Making the additional assumption that $h_b \ll r_{eS}^e$ gives

$$\tilde{\gamma}_{ib}^i - \gamma_{ib}^i \approx -2 \frac{(\tilde{h}_b - h_b)}{r_{eS}^e(\hat{L}_b)} g_0(\hat{L}_b) \hat{\mathbf{u}}_D^i \quad (12.32)$$

where g_0 is the surface gravity and $\hat{\mathbf{u}}_D^i$ is the down unit vector of the local navigation frame, expressed in the ECI frame.³ Transforming from curvilinear position to ECI-frame Cartesian position using (2.71) and (2.95) gives

$$\tilde{\gamma}_{ib}^i - \gamma_{ib}^i \approx \frac{2g_0(\hat{L}_b)}{r_{eS}^e(\hat{L}_b)} \frac{\hat{\mathbf{r}}_{ib}^i}{|\hat{\mathbf{r}}_{ib}^i|^2} \hat{\mathbf{r}}_{ib}^{i\ T} \delta\mathbf{r}_{ib}^i \quad (12.33)$$

Substituting (12.30) and (12.31) into (12.29), and (12.29) and (12.33) into (12.27) gives the time derivative of the velocity error in terms of the Kalman filter states:

$$\delta\mathbf{v}_{ib}^i \approx -(\hat{\mathbf{C}}_{ib}^i \hat{\mathbf{f}}_{ib}^b) \wedge \delta\boldsymbol{\psi}_{ib}^i + \frac{2g_0(\hat{L}_b)}{r_{eS}^e(\hat{L}_b)} \frac{\hat{\mathbf{r}}_{ib}^i}{|\hat{\mathbf{r}}_{ib}^i|^2} \hat{\mathbf{r}}_{ib}^{i\ T} \delta\mathbf{r}_{ib}^i + \hat{\mathbf{C}}_b^i \mathbf{b}_{as} \quad (12.34)$$

12.2.2.3 Position Error Propagation

The time derivative of ECI-frame position is simply velocity, as seen in (5.15):

$$\dot{\mathbf{r}}_{ib}^i = \mathbf{v}_{ib}^i$$

Thus, the time derivative of the position error is the velocity error:³

$$\delta\dot{\mathbf{r}}_{ib}^i = \delta\mathbf{v}_{ib}^i \quad (12.35)$$

12.2.2.4 System and Transition Matrices

The accelerometer and gyro biases, \mathbf{b}_a and \mathbf{b}_g , are assumed not to have a known time variation. Thus,

$$\dot{\mathbf{b}}_a = 0 \quad \dot{\mathbf{b}}_g = 0 \quad (12.36)$$

where estimated, scale factor, cross-coupling, and g-dependent error states are also modeled as nontime-varying. Where separate dynamic bias states are estimates, they should be modeled as exponentially correlated fixed-variance first-order Markov processes using (3.3).

Substituting (12.25) and (12.34) to (12.36) into (3.18), the system matrix, expressed in terms of 3×3 submatrices corresponding to the components of the state vector in (12.17), is

$$\mathbf{F}_{INS}^i = \begin{pmatrix} 0_3 & 0_3 & 0_3 & 0_3 & \hat{\mathbf{C}}_b^i \\ \mathbf{F}_{21}^i & 0_3 & \mathbf{F}_{23}^i & \hat{\mathbf{C}}_b^i & 0_3 \\ 0_3 & \mathbf{I}_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \end{pmatrix} \quad (12.37)$$

where

$$\mathbf{F}_{21}^i = [-(\hat{\mathbf{C}}_b^i \hat{\mathbf{f}}_{ib}^b) \wedge], \quad \mathbf{F}_{23}^i = \frac{2g_0(\hat{\mathbf{L}}_b)}{r_{eS}^e(\hat{\mathbf{L}}_b)} \frac{\hat{\mathbf{r}}_{ib}^i}{|\hat{\mathbf{r}}_{ib}^i|^2} \hat{\mathbf{r}}_{ib}^i {}^T \quad (12.38)$$

In determining where to truncate the expansion of the transition matrix, Φ , in terms of $\mathbf{F}\tau_s$, the maximum magnitude of each higher order term should be estimated by the Kalman filter designer and a determination made as to whether this will have a significant effect on the integration algorithm performance. The truncation does not have to be uniform, so for example, some second-order terms may be neglected whereas others are included. In practice, the state propagation interval, τ_s , needs to be kept down to 1 second or lower; otherwise, the power-series expansion of Φ in terms of $\mathbf{F}\tau_s$ may experience convergence problems.³ Note

that where there is a long interval between measurement updates, such as in many loosely coupled implementations, it may be necessary to run the system propagation at shorter intervals to ensure that the power-series expansion converges.

To first order in $F\tau_s$, the transition matrix is¹

$$\Phi_{INS}^i \approx \begin{bmatrix} I_3 & 0_3 & 0_3 & 0_3 & \hat{C}_b^i \tau_s \\ F_{21}^i \tau_s & I_3 & F_{23}^i \tau_s & \hat{C}_b^i \tau_s & 0_3 \\ 0_3 & I_3 \tau_s & I_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \quad (12.39)$$

To third order in $F\tau$, the transition matrix is

$$\Phi_{INS}^i \approx \begin{bmatrix} I_3 & 0_3 & 0_3 & 0_3 & \hat{C}_b^i \tau_s \\ \Phi_{21}^i & \Phi_{22}^i & \Phi_{23}^i & \Phi_{24}^i & \Phi_{25}^i \\ \Phi_{31}^i & \Phi_{32}^i & \Phi_{33}^i & \Phi_{34}^i & \Phi_{35}^i \\ 0_3 & 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \quad (12.40)$$

where²

$$\begin{aligned} \Phi_{21}^i &= F_{21}^i \tau_s + \frac{1}{6} F_{23}^i F_{21}^i \tau_s^2, & \Phi_{31}^i &= \frac{1}{2} F_{21}^i \tau_s^2 \\ \Phi_{22}^i &= I_3 + \frac{1}{2} F_{23}^i \tau_s^2, & \Phi_{32}^i &= I_3 \tau_s + \frac{1}{6} F_{23}^i \tau_s^3 \\ \Phi_{23}^i &= F_{23}^i \tau_s + \frac{1}{6} F_{23}^{i,2} \tau_s^3, & \Phi_{33}^i &= I_3 + \frac{1}{2} F_{23}^i \tau_s^2 \\ \Phi_{24}^i &= \hat{C}_b^i \tau_s + \frac{1}{6} F_{23}^i \hat{C}_b^i \tau_s^3, & \Phi_{34}^i &= \frac{1}{2} \hat{C}_b^i \tau_s^2 \\ \Phi_{25}^i &= \frac{1}{2} F_{21}^i \hat{C}_b^i \tau_s^2, & \Phi_{35}^i &= \frac{1}{6} F_{21}^i \hat{C}_b^i \tau_s^3 \end{aligned} \quad (12.41)$$

12.2.3 INS State Propagation in the Earth Frame

Where the Kalman-filter-estimated attitude, velocity, and position errors are referenced to and resolved in the ECEF frame, the state vector becomes

$$\mathbf{x}_{INS}^e = \begin{pmatrix} \delta\psi_{eb}^e \\ \delta\mathbf{v}_{eb}^e \\ \delta\mathbf{r}_{eb}^e \\ \mathbf{b}_a \\ \mathbf{b}_g \end{pmatrix} \quad (12.42)$$

where the superscript e is used to denote the ECEF-frame implementation.

As shown in Section 5.2.1, the attitude propagation depends on the Earth rate as well as the gyro measurements. Determination of the attitude error derivative follows its ECI-frame counterpart from (12.18) to (12.24), giving

$$\delta\dot{\psi}_{eb}^e \approx \hat{\mathbf{C}}_b^e (\tilde{\boldsymbol{\omega}}_{eb}^b - \boldsymbol{\omega}_{eb}^b) \quad (12.43)$$

Splitting this up into gyro measurement and Earth-rate terms, and then applying (5.95) and (5.97),

$$\begin{aligned} \delta\dot{\psi}_{eb}^e &\approx \hat{\mathbf{C}}_b^e \delta\boldsymbol{\omega}_{ib}^b - \hat{\mathbf{C}}_b^e (\tilde{\mathbf{C}}_e^b - \mathbf{C}_e^b) \boldsymbol{\omega}_{ie}^e \\ &= \hat{\mathbf{C}}_b^e \delta\boldsymbol{\omega}_{ib}^b - \boldsymbol{\Omega}_{ie}^e \delta\psi_{eb}^e \end{aligned} \quad (12.44)$$

In terms of the Kalman filter states, the rate of change of the attitude error is thus

$$\delta\dot{\psi}_{eb}^e \approx -\boldsymbol{\Omega}_{ie}^e \delta\psi_{eb}^e + \hat{\mathbf{C}}_b^e \mathbf{b}_g \quad (12.45)$$

The rate of change of the Earth-referenced velocity is given by (5.28). Compared to the ECI-frame implementation, this replaces the gravitational term with a gravity term and adds a Coriolis term.

By analogy with (12.34), the time derivative of the velocity error is

$$\delta\dot{\mathbf{v}}_{eb}^e \approx -(\hat{\mathbf{C}}_b^e \hat{\mathbf{f}}_{ib}^b) \wedge \delta\psi_{eb}^e - 2\boldsymbol{\Omega}_{ie}^e \delta\mathbf{v}_{eb}^e + \frac{2g_0(\hat{L}_b)}{r_{es}^e(\hat{L}_b)} \frac{\hat{\mathbf{r}}_{eb}^e}{|\hat{\mathbf{r}}_{eb}^e|^2} \hat{\mathbf{r}}_{eb}^e \text{T} \delta\mathbf{r}_{eb}^e + \hat{\mathbf{C}}_b^e \mathbf{b}_a \quad (12.46)$$

The time derivative of the position error is the same as for the ECI-frame implementation. Thus,

$$\delta\dot{\mathbf{r}}_{eb}^e = \delta\mathbf{v}_{eb}^e \quad (12.47)$$

Substituting (12.45) to (12.47) and (12.36) into (3.18), the system matrix is

$$\mathbf{F}_{INS}^e = \begin{pmatrix} -\Omega_{ie}^e & 0_3 & 0_3 & 0_3 & \hat{\mathbf{C}}_b^e \\ \mathbf{F}_{21}^e & -2\Omega_{ie}^e & \mathbf{F}_{23}^e & \hat{\mathbf{C}}_b^e & 0_3 \\ 0_3 & \mathbf{I}_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \end{pmatrix} \quad (12.48)$$

where

$$\mathbf{F}_{21}^e = [-(\hat{\mathbf{C}}_b^e \hat{\mathbf{f}}_{ib}^b) \wedge], \quad \mathbf{F}_{23}^e = \frac{2g_0(\hat{\mathbf{L}}_b)}{r_{es}^e(\hat{\mathbf{L}}_b)} \frac{\hat{\mathbf{r}}_{eb}^e}{|\hat{\mathbf{r}}_{eb}^e|^2} \hat{\mathbf{r}}_{eb}^e \text{T} \quad (12.49)$$

12.2.4 INS State Propagation Resolved in the Local Navigation Frame

Where the Kalman filter estimated attitude and velocity are Earth-referenced and resolved in the local navigation frame while the position error is expressed in terms of the latitude, longitude, and height, the state vector becomes

$$\mathbf{x}_{INS}^n = \begin{pmatrix} \delta\psi_{nb}^n \\ \delta\mathbf{v}_{eb}^n \\ \delta\mathbf{p}_b \\ \mathbf{b}_a \\ \mathbf{b}_g \end{pmatrix}, \quad \delta\mathbf{p}_b = \begin{pmatrix} \delta L_b \\ \delta \lambda_b \\ \delta h_b \end{pmatrix} \quad (12.50)$$

where the superscript n denotes the local-navigation-frame implementation.

As shown in Section 5.3.1, the attitude propagation equations incorporate a transport-rate term in addition to the Earth-rate and gyro-measurement terms. Following its ECI-frame counterpart from (12.18) to (12.24), the attitude error derivative in the local navigation frame is

$$\dot{\delta\psi}_{nb}^n \approx \hat{\mathbf{C}}_b^n (\tilde{\boldsymbol{\omega}}_{nb}^b - \boldsymbol{\omega}_{nb}^b) \quad (12.51)$$

Expanding this into gyro-measurement, Earth-rate, and transport-rate terms,

$$\dot{\delta\psi}_{nb}^n \approx \hat{\mathbf{C}}_b^n \delta\boldsymbol{\omega}_{ib}^b - \hat{\mathbf{C}}_b^n (\tilde{\boldsymbol{\omega}}_{ie}^b - \boldsymbol{\omega}_{ie}^b) - \hat{\mathbf{C}}_b^n (\tilde{\boldsymbol{\omega}}_{en}^b - \boldsymbol{\omega}_{en}^b) \quad (12.52)$$

Expanding the Earth-rate and transport-rate terms, neglecting products of error states,

$$\begin{aligned} \hat{\mathbf{C}}_b^n (\tilde{\boldsymbol{\omega}}_{ie}^b - \boldsymbol{\omega}_{ie}^b) + \hat{\mathbf{C}}_b^n (\tilde{\boldsymbol{\omega}}_{en}^b - \boldsymbol{\omega}_{en}^b) &\approx \hat{\mathbf{C}}_b^n (\tilde{\mathbf{C}}_n^b - \mathbf{C}_n^b) (\hat{\boldsymbol{\omega}}_{ie}^n - \hat{\boldsymbol{\omega}}_{en}^n) + (\tilde{\boldsymbol{\omega}}_{ie}^n - \boldsymbol{\omega}_{ie}^n) + (\tilde{\boldsymbol{\omega}}_{en}^n - \boldsymbol{\omega}_{en}^n) \\ &\approx \boldsymbol{\Omega}_{in}^n \delta\psi_{nb}^n + (\tilde{\boldsymbol{\omega}}_{ie}^n - \boldsymbol{\omega}_{ie}^n) + (\tilde{\boldsymbol{\omega}}_{en}^n - \boldsymbol{\omega}_{en}^n) \end{aligned} \quad (12.53)$$

From (2.75),

$$\tilde{\boldsymbol{\omega}}_{ie}^n - \boldsymbol{\omega}_{ie}^n = -\omega_{ie} \begin{pmatrix} \sin L_b \\ 0 \\ \cos L_b \end{pmatrix} \delta L_b \quad (12.54)$$

From (5.37), neglecting products of error states and the variation of the radii of curvature with latitude,

$$\begin{aligned} \tilde{\boldsymbol{\omega}}_{en}^n - \boldsymbol{\omega}_{en}^n &\approx \begin{bmatrix} \delta v_{eb,E}^n / (R_E(\hat{L}_b) + \hat{h}_b) \\ -\delta v_{eb,N}^n / (R_N(\hat{L}_b) + \hat{h}_b) \\ -\delta v_{eb,E}^n \tan \hat{L}_b / (R_E(\hat{L}_b) + \hat{h}_b) \end{bmatrix} - \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{\hat{v}_{eb,E}^n}{(R_E(\hat{L}_b) + \hat{h}_b) \cos^2 \hat{L}_b} \delta L_b \\ &+ \begin{bmatrix} -\hat{v}_{eb,E}^n / (R_E(\hat{L}_b) + \hat{h}_b)^2 \\ \hat{v}_{eb,N}^n / (R_N(\hat{L}_b) + \hat{h}_b)^2 \\ \hat{v}_{eb,E}^n \tan \hat{L}_b / (R_E(\hat{L}_b) + \hat{h}_b)^2 \end{bmatrix} \delta h_b \end{aligned} \quad (12.55)$$

The rate of change of the velocity, \mathbf{v}_{eb}^n , is given by (5.46), adding a transport-rate term to the ECEF-frame equivalent. With the height error estimated directly, the gravity term is simplified. Thus, by analogy with (12.34), the time derivative of the velocity error is

$$\begin{aligned} \delta \dot{\mathbf{v}}_{eb}^n &\approx -(\hat{\mathbf{C}}_b^n \hat{\mathbf{f}}_{ib}^b) \wedge \delta \boldsymbol{\psi}_{nb}^n - (\boldsymbol{\Omega}_{en}^n + 2\boldsymbol{\Omega}_{ie}^n) \delta \mathbf{v}_{eb}^n + \mathbf{v}_{eb}^n \wedge (\tilde{\boldsymbol{\omega}}_{en}^n - \boldsymbol{\omega}_{en}^n) \\ &+ 2\mathbf{v}_{eb}^n \wedge (\tilde{\boldsymbol{\omega}}_{ie}^n - \boldsymbol{\omega}_{ie}^n) - \frac{2g_0(\hat{L}_b)}{r_{eS}^e(\hat{L}_b)} \delta h_b + \hat{\mathbf{C}}_b^n \mathbf{b}_a \end{aligned} \quad (12.56)$$

From (2.72), the time derivative of the position error, neglecting products of error states and the variation of the radii of curvature with latitude, is

$$\begin{aligned} \delta \dot{L}_b &= \frac{\delta v_{eb,N}^n}{R_N(\hat{L}_b) + \hat{h}_b} - \frac{\hat{v}_{eb,N}^n \delta h_b}{(R_N(\hat{L}_b) + \hat{h}_b)^2} \\ \delta \dot{\lambda}_b &= \frac{\delta v_{eb,E}^n}{(R_E(\hat{L}_b) + \hat{h}_b) \cos \hat{L}_b} + \frac{\hat{v}_{eb,E}^n \sin \hat{L}_b \delta L_b}{(R_E(\hat{L}_b) + \hat{h}_b) \cos^2 \hat{L}_b} - \frac{\hat{v}_{eb,E}^n \delta h_b}{(R_E(\hat{L}_b) + \hat{h}_b)^2 \cos \hat{L}_b} \\ \delta \dot{h}_b &= -\delta v_{eb,D}^n \end{aligned} \quad (12.57)$$

Substituting (12.52) to (12.57) and (12.36) into (3.18), the system matrix is

$$\mathbf{F}_{INS}^n = \begin{pmatrix} \mathbf{F}_{11}^n & \mathbf{F}_{12}^n & \mathbf{F}_{13}^n & \mathbf{0}_3 & \hat{\mathbf{C}}_b^n \\ \mathbf{F}_{21}^n & \mathbf{F}_{22}^n & \mathbf{F}_{23}^n & \hat{\mathbf{C}}_b^n & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{F}_{32}^n & \mathbf{F}_{33}^n & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{pmatrix} \quad (12.58)$$

where

$$\mathbf{F}_{11}^n = -[\hat{\boldsymbol{\omega}}_{in}^n \wedge] \quad (12.59)$$

$$\mathbf{F}_{12}^n = \begin{bmatrix} 0 & \frac{-1}{R_E(\hat{L}_b) + \hat{h}_b} & 0 \\ \frac{1}{R_N(\hat{L}_b) + \hat{h}_b} & 0 & 0 \\ 0 & \frac{\tan \hat{L}_b}{R_E(\hat{L}_b) + \hat{h}_b} & 0 \end{bmatrix} \quad (12.60)$$

$$\mathbf{F}_{13}^n = \begin{bmatrix} \omega_{ie} \sin \hat{L}_b & 0 & \frac{\hat{v}_{eb,E}^n}{(R_E(\hat{L}_b) + \hat{h}_b)^2} \\ 0 & 0 & \frac{-\hat{v}_{eb,N}^n}{(R_N(\hat{L}_b) + \hat{h}_b)^2} \\ \omega_{ie} \cos \hat{L}_b + \frac{\hat{v}_{eb,E}^n}{(R_E(\hat{L}_b) + \hat{h}_b) \cos^2 \hat{L}_b} & 0 & \frac{-\hat{v}_{eb,E}^n \tan \hat{L}_b}{(R_E(\hat{L}_b) + \hat{h}_b)^2} \end{bmatrix} \quad (12.61)$$

$$\mathbf{F}_{21}^n = -[\hat{\mathbf{C}}_b^n \hat{\mathbf{f}}_{ib}^b \wedge] \quad (12.62)$$

$$\mathbf{F}_{22}^n = \begin{bmatrix} \frac{\hat{v}_{eb,D}^n}{R_N(\hat{L}_b) + \hat{h}_b} & \left(-\frac{2\hat{v}_{eb,E}^n \tan \hat{L}_b}{R_E(\hat{L}_b) + \hat{h}_b} - 2\omega_{ie} \sin \hat{L}_b \right) & \frac{\hat{v}_{eb,N}^n}{R_N(\hat{L}_b) + \hat{h}_b} \\ \left(\frac{\hat{v}_{eb,E}^n \tan \hat{L}_b}{R_E(\hat{L}_b) + \hat{h}_b} + 2\omega_{ie} \sin \hat{L}_b \right) & \frac{\hat{v}_{eb,N}^n \tan \hat{L}_b + \hat{v}_{eb,D}^n}{R_E(\hat{L}_b) + \hat{h}_b} & \left(\frac{\hat{v}_{eb,E}^n}{R_E(\hat{L}_b) + \hat{h}_b} + 2\omega_{ie} \cos \hat{L}_b \right) \\ -\frac{2\hat{v}_{eb,N}^n}{R_N(\hat{L}_b) + \hat{h}_b} & \left(-\frac{2\hat{v}_{eb,E}^n}{R_E(\hat{L}_b) + \hat{h}_b} - 2\omega_{ie} \cos \hat{L}_b \right) & 0 \end{bmatrix} \quad (12.63)$$

$$\mathbf{F}_{23}^n = \begin{bmatrix} \left(-\frac{(\hat{v}_{eb,E}^n)^2 \sec^2 \hat{L}_b}{R_E(\hat{L}_b) + \hat{h}_b} - 2\hat{v}_{eb,E}^n \omega_{ie} \cos \hat{L}_b \right) & 0 & \frac{(\hat{v}_{eb,E}^n)^2 \tan \hat{L}_b}{(R_E(\hat{L}_b) + \hat{h}_b)^2} - \frac{\hat{v}_{eb,N}^n \hat{v}_{eb,D}^n}{(R_N(\hat{L}_b) + \hat{h}_b)^2} \\ \left(\frac{\hat{v}_{eb,N}^n \hat{v}_{eb,E}^n \sec^2 \hat{L}_b}{R_E(\hat{L}_b) + \hat{h}_b} + 2\hat{v}_{eb,N}^n \omega_{ie} \cos \hat{L}_b \right) & 0 & -\frac{\hat{v}_{eb,N}^n \hat{v}_{eb,E}^n \tan \hat{L}_b + \hat{v}_{eb,E}^n \hat{v}_{eb,D}^n}{(R_E(\hat{L}_b) + \hat{h}_b)^2} \\ 2\hat{v}_{eb,E}^n \omega_{ie} \sin \hat{L}_b & 0 & \left(\frac{(\hat{v}_{eb,E}^n)^2}{(R_E(\hat{L}_b) + \hat{h}_b)^2} + \frac{(\hat{v}_{eb,N}^n)^2}{(R_N(\hat{L}_b) + \hat{h}_b)^2} - \frac{2g_0(\hat{L}_b)}{r_{eS}^e(\hat{L}_b)} \right) \end{bmatrix} \quad (12.64)$$

$$\mathbf{F}_{32}^n = \begin{bmatrix} \frac{1}{R_N(\hat{L}_b) + \hat{h}_b} & 0 & 0 \\ 0 & \frac{1}{(R_E(\hat{L}_b) + \hat{h}_b) \cos \hat{L}_b} & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (12.65)$$

$$\mathbf{F}_{33}^n = \begin{bmatrix} 0 & 0 & -\frac{\hat{v}_{eb,N}^n}{(R_N(\hat{L}_b) + \hat{h}_b)^2} \\ \frac{\hat{v}_{eb,E}^n \sin \hat{L}_b}{(R_E(\hat{L}_b) + \hat{h}_b) \cos^2 \hat{L}_b} & 0 & \frac{\hat{v}_{eb,E}^n}{(R_E(\hat{L}_b) + \hat{h}_b)^2 \cos \hat{L}_b} \\ 0 & 0 & 0 \end{bmatrix} \quad (12.66)$$

12.2.5 INS System Noise

The main sources of system noise on the inertial navigation solution are random walk of the velocity error due to noise on the accelerometer specific-force measurements and random walk of the attitude error due to noise on the gyro angular-rate measurements. IMU random noise is discussed in Section 4.4.3. In addition, where separate accelerometer and gyro dynamic bias states are not estimated, the in-run variation of the accelerometer and gyro biases (see Section 4.4.1) can be approximated as white noise. Thus, the INS system noise covariance matrix, \mathbf{Q}_{INS} , assuming 15 states are estimated as defined by (12.17), (12.41), or (12.50), is

$$\mathbf{Q}_{INS} = \begin{pmatrix} n_{rg}^2 \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & n_{ra}^2 \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & n_{bad}^2 \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & n_{bgd}^2 \mathbf{I}_3 \end{pmatrix} \tau_s \quad (12.67)$$

where n_{rg}^2 , n_{ra}^2 , n_{bad}^2 , and n_{bgd}^2 are the power spectral densities of, respectively, the gyro random noise, accelerometer random noise, accelerometer bias variation, and gyro bias variation, and it is assumed that all gyros and all accelerometers have equal noise characteristics.

If σ_{ra} is the standard deviation of the noise on the accelerometer specific-force measurements and σ_{rg} is the standard deviation of the noise on the gyro angular-rate measurements, then the PSDs of the accelerometer and gyro noise are

$$n_{ra}^2 = \sigma_{ra}^2 \tau_i, n_{rg}^2 = \sigma_{rg}^2 \tau_i \quad (12.68)$$

where τ_i is the interval between the input of successive accelerometer and gyro outputs to the inertial navigation equations. Similarly, the bias variation PSDs are

$$n_{bad}^2 = \sigma_{bad}^2 \tau_{bad}, \quad n_{bgd}^2 = \sigma_{bgd}^2 \tau_{bgd} \quad (12.69)$$

where σ_{bad} and σ_{bgd} are the standard deviations of the accelerometer and gyro dynamic biases, respectively, and τ_{bad} and τ_{bgd} are their correlation times.

Where errors such as the accelerometer and gyro scale factor, cross-coupling, and g-dependent errors are not modeled either as states or correlated system noise, their effects may be rather crudely approximated by increasing the modeled accelerometer and gyro random walk. To maintain Kalman filter stability, these white noise approximations must overbound the true impact on the Kalman filter states. Where these errors are estimated as states, a certain level of system noise on the states should be modeled to account for any in-run variation.

The system noise variance for a state, x_{mi} , modeled as an exponentially correlated fixed-variance first-order Markov process, is

$$Q_{mi} = [1 - \exp(-2\tau_s/\tau_{mi})] \sigma_{mi}^2 \quad (12.70)$$

where σ_{mi} is the standard deviation of the state, τ_{mi} is the correlation time, and covariance propagation using (3.11) is assumed. This is applicable to estimation of the accelerometer and gyro dynamic biases.

Advanced system noise modeling is discussed in Section 12.4.4.

12.2.6 GNSS State Propagation and System Noise

Where the receiver clock offset and drift are estimated, the state dynamics are given by (7.188). Where the clock g-dependent error coefficients, s_{cg} , are also estimated, the clock offset varies as

$$\frac{\partial}{\partial t} \delta\rho_{rc} = \delta\dot{\rho}_{rc} + \hat{\mathbf{f}}_{ib}^b \mathbf{s}_{cg}^T \quad (12.71)$$

where it is assumed that the axes of the receiver's reference oscillator are fixed with respect to the IMU. System noise should be modeled on the clock-drift state as described in Section 7.5.2.2, while the g-dependent error coefficients may be assumed constant. The noise on the clock offset is simply the integral of the clock-drift noise, so no system noise is modeled on this state directly.

Ionosphere and range-bias states should be modeled as exponentially correlated fixed-variance first-order Markov processes as described by (3.3), with the system noise given by (12.70). The standard deviation of ionosphere states should be modeled as a function of satellite elevation angle and local time, as discussed in Section 7.4.2. A correlation time of around 30 minutes is suitable. An elevation-dependent standard deviation is also suitable for the range-bias states, while the choice of correlation time depends on whether they are intended primarily to capture the ephemeris, satellite clock, ionosphere, and troposphere errors, demanding a

long correlation time, or the multipath errors, demanding a much smaller correlation time. Where both types of errors are important, separate states can be used.

Where the carrier phase offsets are estimated in coherent deep integration, the system model depends on how the frequency-tracking errors are modeled. Where they are estimated, but not used to generate the NCO commands, the carrier phase offsets are modeled as

$$\delta\dot{\phi}_{ca,j} = 2\pi\delta f_{ca,j} \quad (12.72)$$

Where the frequency-tracking errors are not modeled or they are used to correct the NCO commands, no deterministic dependency of the carrier phase offsets on the other states is then modeled. System noise should be modeled on both the carrier-phase and carrier-frequency states.

12.3 Measurement Models

The measurement model of a Kalman filter is described in Section 3.2.5. In INS/GNSS integration, the differences between measurements output by the GNSS user equipment and predictions of those measurements from the inertial navigation solution are used to update the state vector. Which measurements are used depends on the integration architecture, so the loosely coupled measurement model is described first, followed by the tightly coupled model and models for deep integration. The section concludes with a discussion of how the attitude and instrument errors are estimated.

Consider a measurement, \tilde{m}_G , output by the GNSS user equipment and a prediction of that measurement, \tilde{m}_I , obtained from the raw inertial navigation solution (and the GNSS navigation data message, where appropriate). Estimates of the errors in these measurements, $\hat{\delta m}_G$ and $\hat{\delta m}_I$, can then be obtained from the Kalman filter state vector. There are then two ways in which these can legitimately be assembled into a Kalman filter measurement, z , and estimate thereof, \hat{z}_G^- . These are

$$z_G = \tilde{m}_G - \tilde{m}_I, \quad \hat{z}_G^- = \hat{\delta m}_G - \hat{\delta m}_I \quad (12.73)$$

and

$$z_G = \tilde{m}_G, \quad \hat{z}_G^- = \tilde{m}_I - \hat{\delta m}_I + \hat{\delta m}_G \quad (12.74)$$

Where closed-loop correction of the INS is implemented, the predicted measurement is obtained from the corrected inertial navigation solution and becomes \hat{m}_I . It may also be convenient to do this in an open-loop architecture, noting that $\hat{m}_I = \tilde{m}_I - \hat{\delta m}_I$. The options for the measurement and its estimate are then

$$z_G = \tilde{m}_G - \hat{m}_I, \quad \hat{z}_G^- = \hat{\delta m}_G \quad (12.75)$$

and

$$z_G = \tilde{m}_G, \quad \hat{z}_G^- = \hat{m}_I + \delta\hat{m}_G \quad (12.76)$$

For tightly coupled integration, an extended Kalman filter (Section 3.4.1) is needed for (12.74) and (12.76), but not generally for (12.73) and (12.75). However, the measurement innovation, δz_G^- , is the same in all cases:

$$\delta z_G^- = \tilde{m}_G - \delta\hat{m}_G - \hat{m}_I \quad (12.77)$$

and may be computed directly. Therefore the distinction between a standard Kalman filter and an EKF is one of semantics rather than implementation. Here, the convention of expressing the measurement innovation directly is adopted. Note that the subscript G is used to distinguish GNSS measurement innovations from other types in multisensor integration (Chapter 14).

As discussed in Section 3.3.4, the INS-derived and GNSS-derived measurement data must have the same time of validity. Otherwise, contributions to the measurement innovations caused by time synchronization errors will corrupt the state estimates. A method for measuring the residual data lag is described in [24].

12.3.1 Loosely Coupled Integration

Loosely coupled INS/GNSS integration uses the GNSS user equipment's position and velocity solution. Therefore, the measurement innovation vector comprises the difference between the GNSS and corrected inertial position and velocity solutions, accounting for the lever arm from the INS to the GNSS antenna, \mathbf{l}_{ba}^b , which is assumed here to be well known. The coordinate frames for the measurement innovation should match those for the state vector. Thus,

$$\delta \mathbf{z}_{G,k}^{i-} = \begin{pmatrix} \hat{\mathbf{r}}_{iaG}^i - \hat{\mathbf{r}}_{ib}^i - \hat{\mathbf{C}}_{ib}^i \mathbf{l}_{ba}^b \\ \hat{\mathbf{v}}_{iaG}^i - \hat{\mathbf{v}}_{ib}^i - \hat{\mathbf{C}}_{ib}^i (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) \end{pmatrix}_k \quad (12.78)$$

$$\delta \mathbf{z}_{G,k}^{e-} = \begin{pmatrix} \hat{\mathbf{r}}_{eaG}^e - \hat{\mathbf{r}}_{eb}^e - \hat{\mathbf{C}}_b^e \mathbf{l}_{ba}^b \\ \hat{\mathbf{v}}_{eaG}^e - \hat{\mathbf{v}}_{eb}^e - \hat{\mathbf{C}}_b^e (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{C}}_b^e \mathbf{l}_{ba}^b \end{pmatrix}_k \quad (12.79)$$

and

$$\delta \mathbf{z}_{G,k}^{n-} = \begin{pmatrix} \hat{\mathbf{p}}_{aG} - \hat{\mathbf{p}}_b - \hat{\mathbf{T}}_r^p \hat{\mathbf{C}}_b^n \mathbf{l}_{ba}^b \\ \hat{\mathbf{v}}_{eaG}^n - \hat{\mathbf{v}}_{eb}^n - \hat{\mathbf{C}}_b^n (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) + (\hat{\boldsymbol{\Omega}}_{ie}^n + \hat{\boldsymbol{\Omega}}_{en}^n) \hat{\mathbf{C}}_b^n \mathbf{l}_{ba}^b \end{pmatrix}_k \quad (12.80)$$

where the subscript k denotes the measurement update iteration; superscripts i , e , and n respectively, denote the ECI, ECEF, and local-navigation-frame implementations; the subscript G denotes GNSS indicated; $\boldsymbol{\Omega}_{ie}^e$, $\boldsymbol{\Omega}_{ie}^n$, and $\boldsymbol{\Omega}_{en}^n$ are given by (5.18), (5.34), and (5.37), respectively; and from (2.107), the Cartesian-to-curvilinear position change transformation matrix, $\hat{\mathbf{T}}_r^p$, is

$$\hat{\mathbf{T}}_r^p = \begin{Bmatrix} \frac{1}{R_N(\hat{L}_b) + \hat{h}_b} & 0 & 0 \\ 0 & \frac{1}{(R_E(\hat{L}_b) + \hat{h}_b) \cos \hat{L}_b} & 0 \\ 0 & 0 & -1 \end{Bmatrix} \quad (12.81)$$

From (5.95) and (5.97),

$$\tilde{\mathbf{C}}_b^\gamma = (\mathbf{I}_3 + [\delta\psi_{\gamma b}^\gamma \wedge]) \mathbf{C}_b^\gamma \quad (12.82)$$

Substituting the attitude error states with their residual, $\delta\delta\psi_{\gamma b}^\gamma$ [see (3.4)],

$$\hat{\mathbf{C}}_b^\gamma = (\mathbf{I}_3 + [\delta\delta\psi_{\gamma b}^\gamma \wedge]) \mathbf{C}_b^\gamma \quad (12.83)$$

Therefore,

$$\hat{\mathbf{C}}_b^\gamma \mathbf{l}_{ba}^b = \mathbf{C}_b^\gamma \mathbf{l}_{ba}^b - [(\mathbf{C}_b^\gamma \mathbf{l}_{ba}^b) \wedge] \delta\delta\psi_{\gamma b}^\gamma \quad (12.84)$$

The measurement matrix is given by (3.56):

$$\mathbf{H}_{G,k} = \frac{\partial \mathbf{h}_G(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x} = \hat{\mathbf{x}}_k^-} = \frac{\partial \mathbf{z}_G(\mathbf{x})}{\partial \mathbf{x}} \Big|_{\mathbf{x} = \hat{\mathbf{x}}_k^-}$$

Taking the measurement vector (12.78) and the state vector (12.17), noting that no GNSS states are estimated in loosely coupled integration, and making use of (12.84), the measurement matrix for ECI-frame loosely coupled INS/GNSS integration is

$$\mathbf{H}_{G,k}^i = \begin{pmatrix} \mathbf{H}_{r1}^i & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{H}_{v1}^i & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{H}_{v5}^i \end{pmatrix}_k \quad (12.85)$$

where

$$\begin{aligned} \mathbf{H}_{r1}^i &= [(\hat{\mathbf{C}}_b^i \mathbf{l}_{ba}^b) \wedge] \\ \mathbf{H}_{v1}^i &= [\{\hat{\mathbf{C}}_b^i (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b)\} \wedge] \\ \mathbf{H}_{v5}^i &= \hat{\mathbf{C}}_b^i [\mathbf{l}_{ba}^b \wedge] \end{aligned} \quad (12.86)$$

noting that the corrected INS-indicated attitude, $\hat{\mathbf{C}}_b^i$, is the best available estimate of the true attitude, \mathbf{C}_b^i .

Similarly, from (12.79), (12.42), and (12.84), the ECEF-frame loosely coupled measurement matrix is

$$\mathbf{H}_{G,k}^e = \begin{pmatrix} \mathbf{H}_{r1}^e & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{H}_{v1}^e & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{H}_{v5}^e \end{pmatrix}_k \quad (12.87)$$

where

$$\begin{aligned} \mathbf{H}_{r1}^e &= \left[(\hat{\mathbf{C}}_b^e \mathbf{l}_{ba}^b) \wedge \right] \\ \mathbf{H}_{v1}^e &= \left[\{ \hat{\mathbf{C}}_b^e (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) - \hat{\boldsymbol{\Omega}}_{ie}^e \hat{\mathbf{C}}_b^e \mathbf{l}_{ba}^b \} \wedge \right] \\ \mathbf{H}_{v5}^e &= \hat{\mathbf{C}}_b^e [\mathbf{l}_{ba}^b \wedge] \end{aligned} \quad (12.88)$$

From (12.80), (12.50), and (12.84), the loosely coupled measurement matrix for the Earth-referenced local-navigation-frame implementation is

$$\mathbf{H}_{G,k}^n = \begin{pmatrix} \mathbf{H}_{r1}^n & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{H}_{v1}^n & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{H}_{v5}^n \end{pmatrix}_k \quad (12.89)$$

where

$$\begin{aligned} \mathbf{H}_{r1}^n &= \left[(\hat{\mathbf{T}}_r^p \hat{\mathbf{C}}_b^n \mathbf{l}_{ba}^b) \wedge \right] \\ \mathbf{H}_{v1}^n &= \left[\{ \hat{\mathbf{C}}_b^n (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) - (\hat{\boldsymbol{\Omega}}_{ie}^n + \hat{\boldsymbol{\Omega}}_{en}^n) \hat{\mathbf{C}}_b^n \mathbf{l}_{ba}^b \} \wedge \right] \\ \mathbf{H}_{v5}^n &= \hat{\mathbf{C}}_b^n [\mathbf{l}_{ba}^b \wedge] \end{aligned} \quad (12.90)$$

In practice, the coupling of the attitude errors and gyro biases into the measurements through the lever arm terms is weak. These states are mainly estimated through the change in the velocity error, as described in Section 12.3.4. Therefore, the measurement matrices can often be approximated to

$$\mathbf{H}_{G,k}^{i/e/n} \approx \begin{pmatrix} \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{pmatrix}_k \quad (12.91)$$

noting that the lever arm terms in the measurement innovations, (12.78)–(12.80), must not be neglected.

Ideally, the measurement noise covariance matrix, \mathbf{R}_G , should be based on the error covariance matrix, \mathbf{P} , of the GNSS navigation filter (Section 7.5.2), enabling the GNSS data to be weighted according to the GNSS user equipment's own level of confidence. However, this information is rarely output in practice. The measurement noise covariance is often assumed to be constant but is better modeled as a function of the measured carrier power to noise density, $\tilde{\tau}/\tilde{n}_0$, satellite signal geometry, and acceleration. Acceleration is relevant because the effects of INS-GNSS time synchronization errors are larger under high dynamics. Where the measurement-update interval is shorter than the correlation time of the noise on the GNSS position and velocity solution, the measurement noise covariance assumed in

the integration Kalman filter should be increased to downweight the measurements. Note that reducing the measurement-update interval can better capture the host-vehicle dynamics, improving the observability of some of the attitude and instrument error states.

12.3.2 Tightly Coupled Integration

Tightly coupled INS/GNSS integration uses the GNSS ranging processor's pseudo-range and pseudo-range-rate measurements, obtained from code and carrier tracking, respectively. The measurement model is thus based on that of the GNSS navigation filter, described in Section 7.5.2.3. The measurement innovation vector comprises the differences between the GNSS measured pseudo-range and pseudo-range rates and values predicted from the corrected inertial navigation solution at the same time of validity, estimated receiver clock offset and drift, and navigation-data-indicated satellite positions and velocities. Thus,

$$\delta \mathbf{z}_{G,k}^- = \begin{pmatrix} \delta \mathbf{z}_{\rho,k}^- \\ \delta \mathbf{z}_{r,k}^- \end{pmatrix}, \quad \delta \mathbf{z}_{\rho,k}^- = (\tilde{\rho}_{C1} - \hat{\rho}_{C1}^-, \tilde{\rho}_{C2} - \hat{\rho}_{C2}^-, \dots, \tilde{\rho}_{Cn} - \hat{\rho}_{Cn}^-)_k$$

$$\delta \mathbf{z}_{r,k}^- = (\tilde{\rho}_{C1} - \hat{\rho}_{C1}^-, \tilde{\rho}_{C2} - \hat{\rho}_{C2}^-, \dots, \tilde{\rho}_{Cn} - \hat{\rho}_{Cn}^-)_k \quad (12.92)$$

The corrected pseudo-range and pseudo-range-rate measurements, $\tilde{\rho}_{Cj}$ and $\tilde{\rho}_{Cj}$, are given by (7.43) and their estimated counterparts, $\hat{\rho}_{Cj}$ and $\hat{\rho}_{Cj}$, are given by (7.199) for an ECI-frame calculation and (7.202) or (7.203) for an ECEF-frame calculation. The position and velocity of the user antenna are obtained from the inertial navigation solution using

$$\hat{\mathbf{r}}_{ia}^i = \hat{\mathbf{r}}_{ib}^i + \hat{\mathbf{C}}_b^i \mathbf{l}_{ba}^b \quad (12.93)$$

$$\hat{\mathbf{v}}_{ia}^i = \hat{\mathbf{v}}_{ib}^i + \hat{\mathbf{C}}_b^i (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b)$$

$$\hat{\mathbf{r}}_{ea}^e = \hat{\mathbf{r}}_{eb}^e + \hat{\mathbf{C}}_b^e \mathbf{l}_{ba}^b \quad (12.94)$$

$$\hat{\mathbf{v}}_{ea}^e = \hat{\mathbf{v}}_{eb}^e + \hat{\mathbf{C}}_b^e (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{C}}_b^e \mathbf{l}_{ba}^b$$

or

$$\hat{\mathbf{r}}_{ea}^e = \begin{pmatrix} (R_E(\hat{L}_b) + \hat{h}_b) \cos \hat{L}_b \cos \hat{\lambda}_b \\ (R_E(\hat{L}_b) + \hat{h}_b) \cos \hat{L}_b \sin \hat{\lambda}_b \\ [(1 - e^2) R_E(\hat{L}_b) + \hat{h}_b] \sin \hat{L}_b \end{pmatrix} + \hat{\mathbf{C}}_n^e \hat{\mathbf{C}}_b^n \mathbf{l}_{ba}^b \quad (12.95)$$

$$\hat{\mathbf{v}}_{ea}^e = \hat{\mathbf{C}}_n^e \hat{\mathbf{v}}_{eb}^e + \hat{\mathbf{C}}_n^e \hat{\mathbf{C}}_b^n (\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{ba}^b) + \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{C}}_n^e \hat{\mathbf{C}}_b^n \mathbf{l}_{ba}^b$$

where $\boldsymbol{\Omega}_{ie}^e$ and $\hat{\mathbf{C}}_n^e$ are given by (5.18) and (2.99), respectively.

For tightly coupled integration, the state vector typically comprises the inertial states, receiver clock offset, and drift. Thus,

$$\mathbf{x}^\gamma = \begin{pmatrix} \mathbf{x}_{INS}^\gamma \\ \delta\rho_{rc} \\ \delta\dot{\rho}_{rc} \end{pmatrix} \quad (12.96)$$

where the inertial state vectors given by (12.17), (12.42), and (12.50) are assumed.

The measurement matrix is given by (3.56) and can be expressed in terms of submatrices as

$$\mathbf{H}_{G,k}^\gamma = \begin{pmatrix} \frac{\partial \mathbf{z}_\rho}{\partial \delta \boldsymbol{\psi}_{\gamma b}^\gamma} & \mathbf{0}_{n,3} & \frac{\partial \mathbf{z}_\rho}{\partial \delta \mathbf{r}_{\gamma b}^\gamma} & \mathbf{0}_{n,3} & \mathbf{0}_{n,3} & \frac{\partial \mathbf{z}_\rho}{\partial \delta \rho_{rc}} & \mathbf{0}_{n,1} \\ \frac{\partial \mathbf{z}_r}{\partial \delta \boldsymbol{\psi}_{\gamma b}^\gamma} & \frac{\partial \mathbf{z}_r}{\partial \delta \mathbf{v}_{\gamma b}^\gamma} & \frac{\partial \mathbf{z}_r}{\partial \delta \mathbf{r}_{\gamma b}^\gamma} & \mathbf{0}_{n,3} & \frac{\partial \mathbf{z}_\rho}{\partial \mathbf{b}_g} & \mathbf{0}_{n,1} & \frac{\partial \mathbf{z}_r}{\partial \delta \dot{\rho}_{rc}} \end{pmatrix}_{\mathbf{x} = \hat{\mathbf{x}}_k} \quad \gamma \in i, e \quad (12.97)$$

or

$$\mathbf{H}_{G,k}^n = \begin{pmatrix} \frac{\partial \mathbf{z}_\rho}{\partial \delta \boldsymbol{\psi}_{nb}^n} & \mathbf{0}_{n,3} & \frac{\partial \mathbf{z}_\rho}{\partial \delta \mathbf{p}_b} & \mathbf{0}_{n,3} & \mathbf{0}_{n,3} & \frac{\partial \mathbf{z}_\rho}{\partial \delta \rho_{rc}} & \mathbf{0}_{n,1} \\ \frac{\partial \mathbf{z}_r}{\partial \delta \boldsymbol{\psi}_{nb}^n} & \frac{\partial \mathbf{z}_r}{\partial \delta \mathbf{v}_{eb}^n} & \frac{\partial \mathbf{z}_r}{\partial \delta \mathbf{p}_b} & \mathbf{0}_{n,3} & \frac{\partial \mathbf{z}_\rho}{\partial \mathbf{b}_g} & \mathbf{0}_{n,1} & \frac{\partial \mathbf{z}_r}{\partial \delta \dot{\rho}_{rc}} \end{pmatrix}_{\mathbf{x} = \hat{\mathbf{x}}_k} \quad (12.98)$$

The differentials may be calculated analytically or numerically by perturbing the state estimates and calculating the change in estimate pseudo-range and pseudo-range rate. The dependence of the measurement innovations on the attitude error and of the pseudo-range-rate measurements on the position and gyro errors is weak, so a common approximation to the analytical solution is

$$\mathbf{H}_{G,k}^\gamma \approx \begin{pmatrix} \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{u}_{as,1}^\gamma {}^T & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{u}_{as,2}^\gamma {}^T & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{u}_{as,n}^\gamma {}^T & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 1 & 0 \\ \hline \mathbf{0}_{1,3} & \mathbf{u}_{as,1}^\gamma {}^T & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 0 & 1 \\ \mathbf{0}_{1,3} & \mathbf{u}_{as,2}^\gamma {}^T & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{1,3} & \mathbf{u}_{as,n}^\gamma {}^T & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & \mathbf{0}_{1,3} & 0 & 1 \end{pmatrix}_{\mathbf{x} = \hat{\mathbf{x}}_k} \quad \gamma \in i, e \quad (12.99)$$

or

$$\mathbf{H}_{G,k}^n \approx \begin{pmatrix} 0_{1,3} & 0_{1,3} & \mathbf{h}_{\rho p,1}^T & 0_{1,3} & 0_{1,3} & 1 & 0 \\ 0_{1,3} & 0_{1,3} & \mathbf{h}_{\rho p,2}^T & 0_{1,3} & 0_{1,3} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0_{1,3} & 0_{1,3} & \mathbf{h}_{\rho p,n}^T & 0_{1,3} & 0_{1,3} & 1 & 0 \\ \hline 0_{1,3} & \mathbf{u}_{as,1}^n & 0_{1,3} & 0_{1,3} & 0_{1,3} & 0 & 1 \\ 0_{1,3} & \mathbf{u}_{as,2}^n & 0_{1,3} & 0_{1,3} & 0_{1,3} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0_{1,3} & \mathbf{u}_{as,n}^n & 0_{1,3} & 0_{1,3} & 0_{1,3} & 0 & 1 \end{pmatrix} \quad (12.100)$$

where

$$\mathbf{h}_{\rho p,j} = \begin{pmatrix} (R_N(\hat{L}_b) + \hat{h}_b) u_{as,j,N}^n \\ (R_E(\hat{L}_b) + \hat{h}_b) \cos \hat{L}_b u_{as,j,E}^n \\ -u_{as,j,D}^n \end{pmatrix} \quad (12.101)$$

The measurement noise covariance, \mathbf{R}_G , accounts for GNSS tracking errors, multipath variations, satellite clock noise, and residual INS-GNSS synchronization errors. It should ideally be modeled as a function of \tilde{c}/\tilde{n}_0 and acceleration, though a constant value is often assumed. The matrix \mathbf{R}_G is diagonal, provided that the pseudo-range measurements are not carrier-smoothed. Where pseudo-range measurement updates are performed at a faster rate than about 2 Hz, it may be necessary to increase \mathbf{R}_G to account for time-correlated tracking noise, depending on the design of the GNSS code-tracking loops.

An alternative to using pseudo-range-rate measurements is to use the delta ranges, which are the changes in ADR. The delta-range measurements are less noisy than the pseudo-range rates where the measurement-update interval is much longer than the carrier tracking-loop correlation time. This applies with a 1s update interval and carrier phase tracking. However, using delta range adds complexity to the Kalman filter, requiring either the use of delayed position error states or the incorporation of back-propagation through the system model into the measurement model [25, 26].

Pseudo-range and pseudo-range-rate or delta-range measurements may also be differenced between satellites [26]. This brings the advantage that the receiver clock errors are cancelled out, reducing the number of Kalman filter states required. However, at least two satellites must be tracked in order for any INS calibration to take place, and the measurement model is more complex, including a nondiagonal measurement noise covariance matrix, \mathbf{R}_G .

Some GNSS receivers periodically implement 1-ms clock corrections, causing jumps of 3×10^5 m in their pseudo-range measurements. Tightly coupled integration algorithms using these receivers must detect the jumps (e.g., by applying a threshold to the measurement innovations and correcting the clock offset estimate, $\delta\hat{\rho}_{rc}$, when the threshold is exceeded).

12.3.3 Deep Integration

A deep, or UTC, integration and tracking algorithm inputs raw accumulated correlator outputs, I_s and Q_s , from the GNSS receiver. Coherent and noncoherent deep integration algorithms process these measurements in different ways.

In noncoherent deep integration, the I_s and Q_s are used to form discriminator measurements as in a conventional GNSS ranging processor, described in Sections 7.3.2 and 7.3.3. Pseudo-range and pseudo-range-rate measurement innovations can then be obtained from the discriminator outputs using

$$\delta z_{pj, k}^- = -\frac{c}{f_{co}} (\tilde{x}_{j, k} - \hat{x}_{j, k}) \quad (12.102)$$

$$\delta z_{rj, k}^- = -\frac{c}{f_{ca}} (\delta \tilde{f}_{caj, k} - \delta \hat{f}_{caj, k})$$

where $\tilde{x}_{j, k}$ is the code discriminator output for channel j at iteration k , $\delta \tilde{f}_{caj, k}$ is the corresponding carrier-frequency discriminator output, and $\hat{x}_{j, k}$ and $\delta \hat{f}_{caj, k}$ are their estimated counterparts. It is assumed that the code discriminator outputs are normalized using $\tilde{\epsilon}/\tilde{n}_0$. Note that the estimated discriminator outputs are zero where there are no significant lags in applying the NCO commands (e.g., where a software receiver is used or NCO commands are applied at the IMU output rate). What constitutes a significant lag depends on the host-vehicle dynamics.

The discriminator calculation is typically iterated at 50 Hz, depending on the navigation-data-message rate, where applicable. However, it is not necessary to perform the Kalman filter measurement update at this rate. The measurement innovations given by (12.102) may be averaged over successive iterations to reduce the processor load. Note that averaging should always be used in preference to undersampling, as it gives better signal to noise. Similarly, where navigation-data wipeoff or a data-free signal is used, coherent integration (summation of the I_s and Q_s) should be performed in preference to averaging the measurement innovations. The measurement-update rate must be set high enough to track the receiver clock noise and capture the host-vehicle dynamics. Following measurement innovation averaging, the Kalman filter measurement update then proceeds in the same manner as for tightly coupled integration.

In theory, noncoherent deep integration can also process carrier phase measurements, obtained from conventional carrier-phase discriminators. However, in practice, coherent deep integration is used.

In coherent deep integration, the differences between the measured I_s and Q_s and their predicted values form the measurement innovations. Thus,

$$\delta \mathbf{z}_{G, k}^- = \begin{pmatrix} \delta \mathbf{z}_{G, 1, k}^- \\ \delta \mathbf{z}_{G, 2, k}^- \\ \vdots \\ \delta \mathbf{z}_{G, n, k}^- \end{pmatrix}, \quad \delta \mathbf{z}_{G, j, k}^- = \begin{pmatrix} \tilde{I}_{Ej} - \hat{I}_{Ej}^- \\ \tilde{I}_{Pj} - \hat{I}_{Pj}^- \\ \tilde{I}_{Lj} - \hat{I}_{Lj}^- \\ \tilde{Q}_{Ej} - \hat{Q}_{Ej}^- \\ \tilde{Q}_{Pj} - \hat{Q}_{Pj}^- \\ \tilde{Q}_{Lj} - \hat{Q}_{Lj}^- \end{pmatrix}_k \quad (12.103)$$

The predicted Is and Qs may be calculated using (7.66) as a function of the measured carrier power to noise density, \tilde{c}/\tilde{n}_0 , estimated code tracking error, $\hat{x}_{j,k}$, reference-signal carrier phase offset, $\delta\phi_{caj,k}$, and frequency offset, $\delta f_{caj,k}$, where estimated. The measurement matrix, $\mathbf{H}_{G,k}$, is obtained by differentiating (7.66) with respect to the Kalman filter states. Note that accurate estimates of the carrier phases are needed to compute both the predicted Is and Qs and the measurement matrices; without them, coherent deep integration cannot operate.

Where navigation-data wipeoff or data-free signals are used, the incorporation of the Is and Qs into the Kalman filter as measurements maintains coherent summation over a varying interval, determined by the Kalman filter gain. This optimizes the signal-to-noise performance (see Section 7.2.4.4). Where there are unknown data bits that are not estimated (see Section 8.3.6), coherent integration is limited to the data-bit intervals, and the integration algorithm must use the sign of one of the Is and Qs to correct for the data bits. Thus,

$$\delta \mathbf{z}_{G,j,k}^- = \begin{bmatrix} \tilde{I}_{Ej}\tilde{D}_{j,k} - \hat{I}_{Ej}^- \\ \tilde{I}_{Pj}\tilde{D}_{j,k} - \hat{I}_{Pj}^- \\ \tilde{I}_{Lj}\tilde{D}_{j,k} - \hat{I}_{Lj}^- \\ \tilde{Q}_{Ej}\tilde{D}_{j,k} - \hat{Q}_{Ej}^- \\ \tilde{Q}_{Pj}\tilde{D}_{j,k} - \hat{Q}_{Pj}^- \\ \tilde{Q}_{Lj}\tilde{D}_{j,k} - \hat{Q}_{Lj}^- \end{bmatrix}_k \quad \tilde{D}_{j,k} = \begin{cases} \frac{\text{sign}(\tilde{I}_{Pj})}{\text{sign}(\hat{I}_{Pj})} & |\tilde{I}_{Pj}| \geq |\tilde{Q}_{Pj}| \\ \frac{\text{sign}(\tilde{Q}_{Pj})}{\text{sign}(\hat{Q}_{Pj})} & |\tilde{I}_{Pj}| < |\tilde{Q}_{Pj}| \end{cases} \quad (12.104)$$

To maintain accurate carrier phase estimates, the Kalman filter measurement update must be performed at least 50 Hz. Combined with the additional Kalman filter states required (see Section 12.2.1), this imposes a much higher processing load than the other integration architectures. A common solution is to break up the Kalman filter into federated filters, as shown in Figure 12.7 [22, 23, 27]. Fast tracking filters for each satellite, known as prefilters, estimate the code-phase, carrier-phase, and carrier-frequency tracking errors and output pseudo-range and pseudo-range-rate or delta-range measurement innovations to an INS/GNSS integration filter. The integration filter typically performs measurement updates at 1 or 2 Hz and is analogous to a noncoherent deep integration filter, estimating the INS and receiver clock errors, and optionally, ionosphere or range-bias states. Where the receiver processes GNSS signals on more than one frequency, all measurements from a given satellite should be input to the same tracking filter, which will incorporate additional states. A federated zero-reset architecture [28] (see Section 14.1.4.3) is implemented for the code-phase and carrier-frequency states in the tracking filters, whereby these states are zeroed when data is output to the integration filter because this leads to corrections being applied to the reference signals generated by the GNSS receiver. Alternatively, the tracking filters may be replaced by batch-processing acquisition and tracking algorithms [29] (see Section 8.3.5).

As with the other integration architectures, the measurement noise covariance, \mathbf{R}_G , must be modeled as a function of \tilde{c}/\tilde{n}_0 and may also be varied as a function

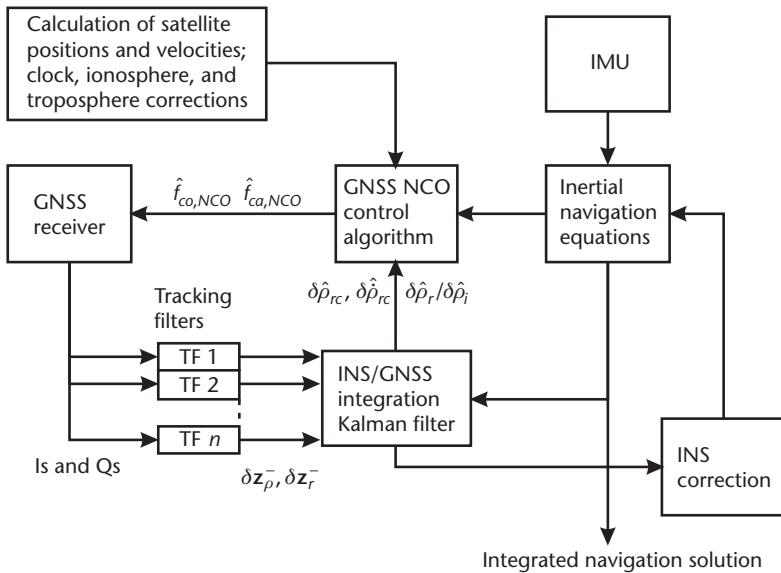


Figure 12.7 Federated coherent deep integration architecture (closed-loop INS correction).

of acceleration. However, for coherent deep integration, R_G must account for noise correlation between the three I measurements and between the three Q measurements from each channel, requiring off-diagonal terms.

Both types of deep integration can bridge short-term GNSS signal outages, so code tracking lock is determined by comparing the pseudo-range uncertainties obtained by resolving the position-error and clock-offset state uncertainties along each user-satellite line of sight, with the code chip length. Where there is insufficient c/n_0 to track code, R_G will be large, causing the state uncertainties to grow. A similar approach is applicable to carrier tracking.

12.3.4 Estimation of Attitude and Instrument Errors

In INS/GNSS integration, the measurement innovations input to the Kalman filter are based on position and velocity or pseudo-range and pseudo-range rate. As (12.91), (12.99), and (12.100) show, the direct coupling of these measurements to the attitude-error and instrument-error states is usually negligible. Yet, the Kalman filter still estimates these errors, and how it does this is not intuitive. Here, an explanation is presented.

As explained in Section 12.2.1, the attitude and instrument errors are observed through the growth in the velocity error they produce, with the attitude errors and accelerometer biases inducing a linear growth in the velocity error and gyro biases inducing a quadratic growth. The corresponding growths in position error are quadratic and cubic, respectively. This coupling of the states is represented by the system model (Sections 12.2.2–12.2.4). Each time the error covariance matrix, P , is propagated through the system model [using (3.11) or (3.32)], information on the correlations between the residual attitude and instrument errors and the residual position and velocity errors is built up in the off-diagonal elements of P . This enables

the measurement model to estimate corrections to the attitude and instrument error estimates from measurements of the position and velocity errors (or linear combinations thereof).

When a measurement update is performed, the error covariance matrix, \mathbf{P} , is used in the calculation of the Kalman gain matrix, \mathbf{K} , using (3.15). The products of the off-diagonal elements of \mathbf{P} with the elements of the measurement matrix, \mathbf{H}_G , coupling the position and velocity errors to the measurements give elements of \mathbf{K} coupling the measurements to the attitude and instrument errors. Thus, when a state-vector update, (3.16), is performed, the attitude and instrument error estimates are updated alongside the position and velocity error estimates and any GNSS states.

12.4 Advanced INS/GNSS Integration

This section collects together a number of advanced INS/GNSS integration topics. Integration of INS with differential, carrier-phase, and multiantenna GNSS, modeling large heading errors, advanced IMU error modeling, and smoothing are discussed.

12.4.1 Differential GNSS

Differential GNSS, described in Section 8.1, improves position accuracy by calibrating out much of the temporally and spatially correlated biases in the pseudo-range measurements due to ephemeris prediction errors and residual satellite clock errors, ionospheric refraction, and sometimes tropospheric refraction.

The architectures for integrating DGNSS with INS are essentially the same as for stand-alone GNSS. Differential corrections are applied to the pseudo-range measurements. In loosely coupled integration, this occurs within the GNSS user equipment. For tightly coupled integration, the pseudo-range measurements may be corrected by the GNSS ranging processor or within the integration algorithm's measurement model. For deep integration, the differential corrections are applied within the NCO control algorithm.

The measurement noise covariance, \mathbf{R}_G , is the same as for stand-alone GNSS, as the tracking noise, multipath, and time-synchronization errors it accounts for are unchanged by the application of differential corrections. However, the standard deviation of the range biases, modeled either as states (see Sections 12.2.1 and 12.2.6) or as additional position and clock-offset uncertainty (see Section 7.5.2.4), should be reduced.

12.4.2 Carrier-Phase Positioning and GNSS Attitude

As described in Section 8.2, real-time centimeter-accuracy positioning can be obtained by comparing GNSS ADR measurements with those made by user equipment at a precisely surveyed base station. Integration with INS, as well as bridging the position solution through GNSS outages, can also bridge the ambiguity resolu-

tion process through outages of up to about a minute [30, 31] and aid the detection and correction of cycle slips [14].

GNSS attitude determination uses relative carrier-phase positioning between antennas mounted on the same vehicle as described in Section 8.2.2. GNSS attitude is very noisy, but does not drift, making it highly complementary to INS attitude and a solution to the heading calibration problem that occurs for some INS/GNSS applications. By combining INS with multiantenna GNSS, a precise and stable attitude solution may be obtained. The inertial attitude solution can also aid the GNSS ambiguity resolution process by significantly reducing the search space. With a short baseline, the inertial attitude may completely resolve the ambiguity. A full GNSS attitude solution requires three or more antennas. However, two antennas is sufficient for INS/GNSS where conventional INS/GNSS meets the roll and pitch accuracy requirements [32].

Loosely coupled integration of carrier-phase GNSS with INS is the same as integration of stand-alone or differential GNSS, except that a smaller measurement noise covariance, \mathbf{R}_G , is modeled as well as smaller range biases. For loosely coupled integration of GNSS attitude, an additional measurement innovation is added:

$$\mathbf{I}_3 + [\delta \mathbf{z}_{\psi, k}^{\gamma-} \wedge] \approx \tilde{\mathbf{C}}_{aG, k}^{\gamma} \mathbf{C}_b^a \hat{\mathbf{C}}_{\gamma, k}^{b-} \quad \gamma \in i, e, n \quad (12.105)$$

where $\tilde{\mathbf{C}}_{aG}^{\gamma}$ is the GNSS attitude measurement, and the relative orientation of the INS and GNSS body frames, \mathbf{C}_b^a , is assumed to be known. The measurement matrix is

$$\mathbf{H}_{\psi k}^{i/e/n} = (-\mathbf{I}_3 \quad \mathbf{0}_3 \quad \mathbf{0}_3 \quad \mathbf{0}_3 \quad \mathbf{0}_3) \quad (12.106)$$

assuming the state vector defined by (12.17), (12.42), or (12.50).

Tightly coupled INS/GNSS integration may be performed independently of ambiguity resolution by inputting corrected carrier-derived pseudo-range measurements [3], noting that there is no benefit in using pseudo-range rate where the pseudo-ranges are carrier derived. Carrier-phase GNSS positioning may be integrated with INS using standard tightly coupled INS/GNSS integration algorithms with a smaller measurement noise covariance, \mathbf{R}_G , and range biases modeled. Alternatively, the carrier-phase measurements, corrected for the integer ambiguity, may be differenced across satellites, differenced between the mobile and base station, or double-differenced across both, provided the noise correlations are correctly accounted for in the measurement noise covariance, \mathbf{R}_G [33]. Note that measurements differenced between the mobile and base station provide a relative rather than absolute position solution.

GNSS attitude is tightly integrated by processing separate code- and carrier-derived pseudo-ranges from each antenna in a single Kalman filter. Note that the appropriate lever arms must be modeled and the attitude-error components of the measurement matrix, \mathbf{H}_G , may not be neglected [3]. Where no base station is used, the measurement noise covariance matrix, \mathbf{R}_G , represents corresponding measurements from different antennas as highly correlated, enabling carrier-

precision attitude and code-precision position and velocity to be derived from the same pseudo-ranges. Measurements may be single-differenced across satellites to eliminate the receiver clock errors. However, differencing measurements across antennas eliminates the position and velocity information, leaving only attitude.

Loosely coupled integration of GNSS attitude measurements with INS may be combined with tightly coupled integration of GNSS range measurements. However, tightly coupled attitude integration is not compatible with loosely coupled position and velocity integration unless carrier-derived pseudo-ranges are differenced across antennas. The accuracy of INS/GNSS attitude determination depends on the quality of the inertial sensors and the antenna separation. Longer lever arms produce more precise GNSS attitude measurements but can be subject to flexure. One solution to the flexure problem is to measure it using the main IMU, additional inertial sensors at the antennas, and/or strain gauges [34].

In both the loosely coupled and tightly coupled integration architectures, the corrected INS position and/or attitude solution may be used to aid GNSS ambiguity resolution. However, both the integration and ambiguity resolution algorithms must be carefully tuned to avoid positive-feedback problems. Alternatively, where the ambiguity resolution algorithm is Kalman filter-based, it may be combined with the integration algorithm into a single algorithm, estimating both the INS errors and the carrier integer ambiguities. This also applies to deep integration.

12.4.3 Large Heading Errors

In the examples of INS/GNSS Kalman filters presented in Sections 12.2 and 12.3, the small angle approximation is applied to the attitude errors. This is usually valid for the roll and pitch attitude, which may be observed through leveling (Section 5.5.2). However, the heading (or azimuth) is more difficult to initialize. Only the higher grades of INS are capable of gyrocompassing, while magnetic heading is subject to environmental anomalies and heading derived from the GNSS trajectory is subject to sideslip-induced errors.

Where the small angle approximation is applied to the attitude components resolved about the horizontal axes, but not the heading, the attitude-error coordinate transformation matrix may be expressed as

$$\begin{aligned} \delta C_b^n &\approx \begin{pmatrix} \cos \delta\psi_{nb,D}^n & -\sin \delta\psi_{nb,D}^n & \delta\psi_{nb,N}^n \sin \delta\psi_{nb,D}^n + \delta\psi_{nb,E}^n \cos \delta\psi_{nb,D}^n \\ \sin \delta\psi_{nb,D}^n & \cos \delta\psi_{nb,D}^n & -\delta\psi_{nb,N}^n \cos \delta\psi_{nb,D}^n + \delta\psi_{nb,E}^n \sin \delta\psi_{nb,D}^n \\ -\delta\psi_{nb,E}^n & \delta\psi_{nb,N}^n & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos \delta\psi_{nb,D}^n & -\sin \delta\psi_{nb,D}^n & 0 \\ \sin \delta\psi_{nb,D}^n & \cos \delta\psi_{nb,D}^n & 0 \\ 0 & 0 & 1 \end{pmatrix} \left(I_3 + \left[\begin{pmatrix} \delta\psi_{nb,N}^n \\ \delta\psi_{nb,E}^n \\ 0 \end{pmatrix} \wedge \right] \right) \end{aligned} \quad (12.107)$$

noting that the down component of the attitude error, $\delta\psi_{nb,D}^n$, is the heading error, $\delta\psi_{nb}$.

Using (12.107), the system model is no longer a linear function of the error states, a key requirement of Kalman filtering. One solution is to replace the heading error state with sine and cosine terms, $\sin \delta\psi_{nb}$ and $\cos \delta\psi_{nb}$ [35] or $\delta\sin \psi_{nb}$ and $\delta\cos \psi_{nb}$ [36]. These enable INS/GNSS integration and other forms of fine alignment (Section 5.5.3) to take place with no prior knowledge of heading.

Where the heading error is very large, the products of the heading error state(s) with other states are no longer negligible, so a system model of at least second order is required. An extended Kalman filter (Section 3.4.1) is not suitable, as it linearizes the propagation of the error covariance matrix, \mathbf{P} , through which the attitude errors are observed (Section 12.3.4). Alignment with large heading errors has been demonstrated using an unscented Kalman filter [37] and a number of other nonlinear filters [38]. However, other research has shown that a conventional approach with closed-loop INS correction works equally well for initial attitude uncertainties of at least 30° [39].

12.4.4 Advanced IMU Error Modeling

Low grades of IMU, particularly those using MEMS sensors, can exhibit high levels of noise. To optimize the Kalman filter gain, as discussed in Section 3.3.1, it is important to match the assumed sensor noise to its true value. Unfortunately, the manufacturer's specification may not be an accurate guide due both to variation in noise performance between individual sensors and to variation in effective noise levels with the vibration environment (see Section 4.4.3).

One solution is to use an adaptive Kalman filter to vary the assumed system noise according to the measurement innovations as described in Section 3.4.3. Both the innovation-based and multiple-model adaptive estimation techniques have been shown to speed up the rate of convergence of the state estimates with their true counterparts [40–43]. Interestingly, these algorithms tend to select high levels of system noise initially, leading to high Kalman filter gains and faster convergence, and then switch to lower system noise after convergence, producing lower gains, more stable state estimates, and smaller state uncertainties.

Where the dominant vibration modes are known, or determined from the IMU data, they can be incorporated into the Kalman filter. Modeling the sensor noise as correlated by second-order Markov processes using a Schmidt-Kalman filter (Section 3.4.2) has been shown to improve alignment performance for missiles in an air-carriage environment [44].

MEMS IMUs can also exhibit complex higher order systematic and slowly time-varying errors that are difficult to model using a Kalman filter. However, these errors can be modeled using an ANN alongside a conventional Kalman filter estimating the standard 15 INS error states (see Sections 12.2.2–12.2.4). The neural network is trained while GNSS data is available and then predicts the residual INS position errors during outages, noting that the dynamics in the training and prediction phases must be similar. Significant improvements in position accuracy during GNSS outages have been demonstrated using a number of hybrid ANN/Kalman filter integration algorithms, compared to a Kalman filter alone [45–47].

Accelerometer and gyro biases (Section 4.4.1) vary over time. This is conventionally accounted for by modeling white system noise on the bias states, or by

modeling each bias state as either first-order Markov processes or the sum of a constant and a Markov state. However, these models are only a rough approximation of the sensor behavior. Performance improvements have been demonstrated by representing the biases as second- and third-order autoregressive models, tuned to each sensor type [48, 49], while a frequency-domain approach [50] avoids the need to make a priori assumptions about the time variations and can improve the speed of convergence.

All of these methods of modeling IMU errors require more processing capacity and were relatively immature at the time of writing. However, they demonstrate the scope to improve upon the conventional system models described in Section 12.2.

12.4.5 Smoothing

For many applications, such as surveying, geo-referencing, vehicle testing, and military ranges, the navigation solution is required for analysis after the event. In these cases, the INS errors can be calibrated using GPS measurements taken after the time of interest as well as before. A standard Kalman filter will not do this; the solution is to use a Kalman smoother, as described in Section 3.4.5. Whether smoothing significantly improves performance depends on the application. It is useful where it is not practical to undergo a period of INS calibration before the data set of interest or where a heading solution is required and the heading error is difficult to observe. However, it has the biggest impact where GNSS signal availability is relatively poor, such as in urban areas, particularly where carrier-phase accuracy is required [51] or a low-grade IMU is used [52]. Smoothing effectively halves the period of INS drift during GNSS outages, reducing the maximum position error by a factor of up to 4.

References

- [1] Titterton, D. H., and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., Stevenage, U.K.: IEE, 2004.
- [2] Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, New York: Wiley, 2001.
- [3] Farrell, J. A., and M. Barth, *The Global Positioning System and Inertial Navigation*, New York: McGraw-Hill, 1999.
- [4] Phillips, R. E., and G. T. Schmidt, “GPS/INS Integration,” *Proc. NATO AGARD MSP Lecture Series on ‘System Implications and Innovative Applications of Satellite Navigation,’ LS-207*, Paris, France, July 1996.
- [5] Greenspan, R. L., “GPS and Inertial Integration,” in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 187–220.
- [6] Groves, P. D., “Principles of Integrated Navigation,” course notes, QinetiQ Ltd., 2002.
- [7] Wagner, J. F., G. Kasties, and M. Klotz, “An Alternative Filter Approach to Integrate Satellite Navigation and Inertial Sensors,” *Proc. ION NTM*, Santa Monica, CA, January 1997, pp. 141–150.
- [8] Cox, D. B., Jr., “Integration of GPS with Inertial Navigation Systems,” *Navigation: JION*, Vol. 25, No. 2, 1978, pp. 236–245.

- [9] Groves, P. D., and D. C. Long, "Inertially-Aided GPS Signal Re-Acquisition in Poor Signal to Noise Environments and Tracking Maintenance Through Short Signal Outages," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 2408–2417.
- [10] Alban, S. et al., "Performance Analysis and Architectures for INS-Aided GPS Tracking Loops," *Proc. ION NTM*, Anaheim, CA, January 2003, pp. 611–622.
- [11] Groves, P. D., and D. C. Long, "Combating GNSS Interference with Advanced Inertial Integration," *Journal of Navigation*, Vol. 58, No. 3, 2005, pp. 419–432.
- [12] Groves, P. D., and D. C. Long, "Adaptive Tightly-Coupled, a Low Cost Alternative Anti-Jam INS/GPS Integration Technique," *Proc. ION NTM*, Anaheim, CA, January 2003, pp. 429–440.
- [13] Bye, C. T., G. L. Hartmann, and A. Killen, "Inertial and GPS Technology Advances on the GGP Program," *Proc ION 53rd AM*, June 1997, pp. 639–648.
- [14] Colombo, O. L., U. V. Bhapkar, and A. G. Evans, "Inertial-Aided Cycle-Slip Detection/Correction for Precise, Long-Baseline Kinematic GPS," *Proc. ION GPS-99*, Nashville, TN, September 1999, pp. 1915–1921.
- [15] Copps, E. M. et al., "Optimal Processing of GPS Signals," *Navigation: JION*, Vol. 27, No. 3, 1980, pp. 171–182.
- [16] Groves, P. D., C. J. Mather, and A. A. Macaulay, "Demonstration of Non-Coherent Deep INS/GPS Integration for Optimized Signal to Noise Performance," *Proc. ION GNSS 2007*, Fort Worth, TX, September 2007.
- [17] Gustafson, D., J. Dowdle, and K. Flueckiger, "A Deeply Integrated Adaptive GPS-Based Navigator with Extended Range Code Tracking," *Proc. IEEE PLANS*, San Diego, CA, March 2000, pp. 118–124.
- [18] Buck, T. M., J. Wilmore, and M. J. Cook, "A High G, MEMS Based, Deeply Integrated, INS/GPS, Guidance, Navigation and Control Flight Management Unit," *Proc. IEEE/ION PLANS*, San Diego, CA, April 2006, pp. 772–794.
- [19] Soloviev, A., S. Gunawardena, and F. Van Graas, "Deeply Integrated GPS/Low-Cost IMU for Low CNR Signal Processing: Flight Test Results and Real Time Implementation," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1598–1608.
- [20] Jun, W., et al., "Quaternion-Based Attitude Estimation using Multiple GPS Antennas, MEMS IMU," *Proc. ION GPS/GNSS 2003*, Portland, OR, September 2003, pp. 480–488.
- [21] Rhee, I., M. F. Abdel-Hafez, and J. L. Speyer, "Observability of an Integrated GPS/INS During Manoeuvres," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 40, No. 2, 2004, pp. 526–535 and 1421.
- [22] Sennott, J. W., and D. Senffner, "Navigation Receiver with Coupled Signal-Tracking Channels," U.S. Patent 5,343,209, granted 1994.
- [23] Abbott, A. S., and W. E. Lillo, "Global Positioning System and Inertial Measuring Unit Ultralight Coupling Method," U.S. Patent 6,516,021, granted 2003.
- [24] Lee, K. H., J. G. Lee, and G.-I. Jee, "Calibration of Measurement Delay in Global Positioning System/Strapdown Inertial Navigation System," *Journal of Guidance, Control and Dynamics*, Vol. 25, No.2, 2002, pp. 240–247.
- [25] Wendel, J., T. Obert, and G. F. Trommer, "Enhancement of a Tightly Coupled GPS/INS System for High Precision Attitude Determination of Land Vehicles," *Proc ION 59th AM*, Albuquerque, NM, June 2003, pp. 200–208.
- [26] Farrell, J. L., "GPS/INS—Streamlined," *Navigation: JION*, Vol. 49, No. 4, 2002, pp. 171–182.
- [27] Beser, J., et al., "Trunav: A Low-Cost Guidance/Navigation Unit Integrating a SAASM-Based GPS and MEMS IMU in a Deeply Coupled Mechanization," *Proc. ION GPS 2002*, Portland OR, September 2002, pp. 545–555.
- [28] Carlson, N. A., "Federated Filter for Distributed Navigation and Tracking Applications," *Proc. ION 58th AM*, Albuquerque, NM, June 2002, pp. 340–353.

- [29] Van Graas, F., et al., "Comparison of Two Approaches for GNSS Receiver Algorithms: Batch Processing and Sequential Processing Considerations," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 200–211.
- [30] Petovello, M. G., M. E. Cannon, and G. Lachapelle, "Benefits of Using a Tactical Grade IMU for High-Accuracy Processing," *Navigation: JION*, Vol. 51, No. 1, 2004, pp. 1–12.
- [31] Zhang, H. T., M. G. Petovello, and M. E. Cannon, "Performance Comparison of Kinematic GPS Integrated with Different Tactical Level IMUs," *Proc. ION NTM*, San Diego, CA, January 2005, pp. 243–254.
- [32] Tazartes, D., et al., "Synergistic Interferometric GPS-INS," *Proc. ION NTM*, Anaheim, CA, January 1995, pp. 657–671.
- [33] Lorga, J. F. M., Q. P. Chu, and J. A. Mulder, "Tightly-Coupled IMU/GPS Carrier-Phase Navigation System," *Proc. ION NTM*, Anaheim, CA, January 2003, pp. 385–396.
- [34] Wagner, J. F., and G. Kasties, "Modelling the Vehicle Kinematics as Key Element for the Design of Integrated Navigation System," *Proc. IAIN World Congress*, Berlin, Germany, October 2003.
- [35] Scherzinger, B. M., "Inertial Navigator Error Models for Large Heading Uncertainty," *Proc. IEEE PLANS*, Atlanta, GA, April 1996, pp. 477–484.
- [36] Rogers, R. M., "IMU In-Motion Alignment Without Benefit of Attitude Initialization," *Navigation: JION*, Vol. 44, No. 3, 1997, pp. 301–311.
- [37] Shin, E.-H., and N. El-Sheimy, "An Unscented Kalman Filter for In-Motion Alignment of Low-Cost IMUs," *Proc. IEEE PLANS*, Monterey, CA, April 2004, pp. 273–279.
- [38] Fujioka, S., et al., "Comparison of Nonlinear Filtering Methods for INS/GPS In-Motion Alignment," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 467–477.
- [39] Wendel, J., et al., "A Performance Comparison of Tightly Coupled GPS/INS Navigation Systems Based on Extended and Sigma Point Kalman Filters," *Navigation: JION*, Vol. 53, No. 1, 2006, pp. 21–31.
- [40] Mohammed, A. H., and K. P. Schwarz, "Adaptive Kalman Filtering for INS/GPS," *Journal of Geodesy*, Vol. 73, 1999, pp. 193–203.
- [41] Wang, J., M. Stewart, and M. Tsakiri, "Online Stochastic Modelling for INS/GPS Integration," *Proc. ION GPS '99*, Nashville, TN, September 1999, pp. 1887–1895.
- [42] Hide, C., T. Moore, and M. Smith, "Adaptive Kalman Filtering for Low Cost INS/GPS," *Journal of Navigation*, Vol. 56, No. 1, 2003, pp. 143–152.
- [43] Hide, C., T. Moore, and M. Smith, "Multiple Model Kalman Filtering for GPS and Low-Cost INS Integration," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1096–1103.
- [44] Wendel, J., and G. F. Trommer, "An Efficient Method for Considering Time Correlated Noise in GPS/INS Integration," *Proc. ION NTM*, San Diego, CA, January 2004, pp. 903–911.
- [45] Kaygisiz, B. H., I. Erkmen, and A. M. Erkmen, "GPS/INS Enhancement for Land Navigation Using Neural Network," *Journal of Navigation*, Vol. 57, No. 2, 2004, pp. 297–310.
- [46] El-Sheimy, N., W. Abdel-Hamid, and G. Lachapelle, "An Adaptive Neuro-Fuzzy Model for Bridging GPS Outages in MEMS-IMU/GPS Land Vehicle Navigation," *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 1088–1095.
- [47] Goodall, C., N. El-Sheimy, and K.-W. Chiang, "The Development of a GPS/MEMS INS Integrated System Utilizing a Hybrid Processing Architecture," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 1444–1455.
- [48] Nassar, S., et al., "Modeling Inertial Sensor Errors Using Autoregressive (AR) Models," *Proc. ION NTM*, Anaheim, CA, January 2003, pp. 116–125.
- [49] Nassar, S., K. P. Schwarz, and N. El-Sheimy, "INS and INS/GPS Accuracy Improvement Using Autoregressive (AR) Modeling of INS Sensor Errors," *Proc. ION NTM*, San Diego, CA, January 2004, pp. 936–944.

- [50] Soloviev, A., and F. van Graas, "Investigation into Performance Characteristics of Frequency Domain INS Calibration Procedure Under Noisy GPS Environments," *Proc. ION GPS 2002*, Portland, OR, September 2002, pp. 1454–1463.
- [51] Shin, E.-H., and N. El-Sheimy, "Optimizing Smoothing Computation for Near-Real-Time GPS Measurement Gap Filling in INS/GPS Systems," *Proc. ION GPS 2002*, Portland, OR, September 2002, pp. 1434–1441.
- [52] Hide, C., and T. Moore, "GPS and Low Cost INS Integration for Positioning in the Urban Environment," *Proc. ION GNSS 2005*, Long Beach, CA, September 2005, pp. 1007–1015.

Selected Bibliography

- Farrell, J. L., *GNSS Aided Navigation and Tracking*, Baltimore, MD: American Literary Press, 2007.
- Rogers, R. M., *Applied Mathematics in Integrated Navigation Systems*, Reston, VA: AIAA, 2000.
- Wendel, J., *Integrierte Navigationssysteme: Sensordatenfusion, GPS und Inertiale Navigation*, München, Deutschland: Oldenbourg Verlag, 2007.

Endnotes

1. This and subsequent paragraphs are based on material written by the author for QinetiQ, so comprise QinetiQ copyright material.
2. End of QinetiQ copyright material.
3. This paragraph, up to this point, is based on material written by the author for QinetiQ, so comprises QinetiQ copyright material.

INS Alignment and Zero Velocity Updates

This chapter describes a number of fine alignment methods for improving the calibration of an INS attitude solution and inertial sensor errors between the initialization of the INS and the use of its navigation solution. It follows on from the discussion of INS initialization and alignment in Section 5.5 and exploits synergies with INS/GNSS integration, described in Chapter 12.

Section 13.1 describes transfer alignment, in which measurements from a nearby reference navigation system are used to initialize, align, and calibrate an INS over a few seconds or minutes while in motion. Section 13.2 describes quasi-stationary alignment without heading initialization. Section 13.3 describes quasi-stationary fine alignment where the heading is known and the related zero velocity update (ZVU or ZUPT), which may be applied wherever an INS is known to be stationary.

13.1 Transfer Alignment

Transfer alignment is used to initialize, align, and calibrate an INS in motion. Navigation applications include guided weapons and UAVs launched from aircraft and ships; aircraft, AUVs, and ROVs launched from ships; and torpedos launched from ships and submarines. The discussion here focuses on airborne transfer alignment, sometimes known as in-flight alignment, a term also applied to alignment from GNSS.

Figure 13.1 illustrates the most challenging airborne transfer alignment environment, whereby the weapon or UAV containing the aligning INS is mounted on a wing pylon and the reference navigation system, an INS or integrated INS/GNSS, is mounted in the aircraft body. This maximizes the linear and angular motion of the lever arm between the two navigation systems. Flexure of both the wing and pylon occurs when the aircraft maneuvers and as the loading changes over time due to fuel consumption and the launching of other weapons or UAVs. Vibration occurs due to turbulence and the transmission of engine vibration from the aircraft. More benign environments include a shoulder pylon, closer to the aircraft body, and a trapeze inside the aircraft, though flexure and vibration still occur in both cases.

Transfer alignment comprises up to three phases: a “one-shot” initialization, a measurement-matching phase, and a reinitialization immediately prior to the launch of the aligning INS’s host vehicle [1]. The one-shot phase initializes the

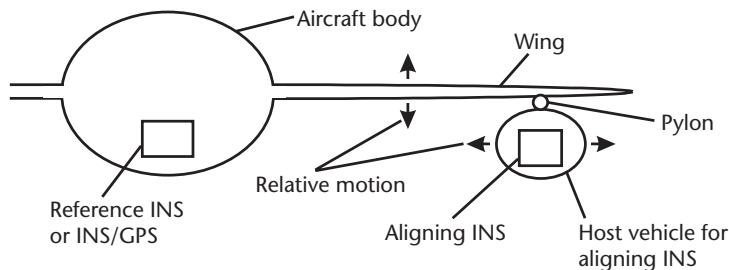


Figure 13.1 Example of an airborne transfer alignment environment.

aligning INS with the reference navigation system's position, velocity, and attitude solution, corrected with the estimated lever arm between and the relative orientation of the two navigation systems, and predicted forward to compensate the data transmission lag. This is good enough for position initialization. However, flexure and vibration can limit the attitude initialization accuracy to about 2° , which, in turn, can lead to position error growth in excess of 500m over the first minute (see Section 5.6.1).

The measurement-matching phase compares the aligning-INS and reference navigation solutions using a Kalman filter (Chapter 3) to estimate corrections to the aligning-INS navigation solution and calibrate the IMU errors. The duration of this phase can vary from 2 seconds to many minutes, depending on the application. Best performance requires at least 2 minutes. Figure 13.2 illustrates this with continuous closed-loop correction of the INS (see Section 12.1.1). Conventional measurement matching uses only linear measurements, while rapid alignment also uses angular measurements. Both are described next, followed by a discussion of different types of reference navigation system.

At the reinitialization phase, the aligning-INS position solution is reset from the aircraft's integrated position solution. Furthermore, if the error covariance matrix, P , is used to initialize a subsequent integration Kalman filter, it must be

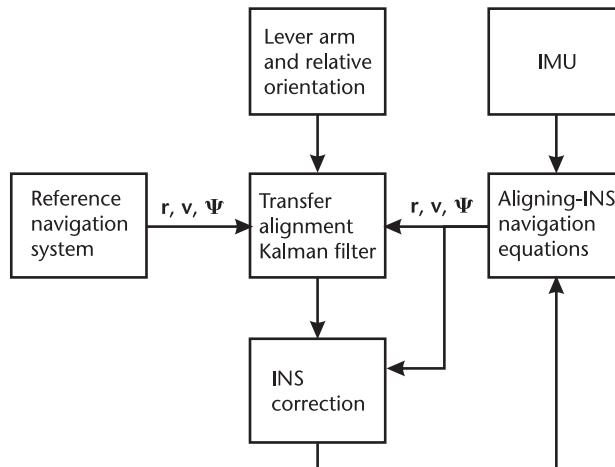


Figure 13.2 Architecture of transfer-alignment measurement-matching phase.

reset to account for the errors of the reference navigation system. This is because these errors are inherited by the aligning INS, but are not accounted for by \mathbf{P} during the measurement-matching phase.

13.1.1 Conventional Measurement Matching

Conventional transfer alignment uses only linear measurement matching. In contrast to many forms of INS/GNSS integration, either position or velocity measurements are used, not both. Position and velocity measurements comprise the same information, as one is the integral of the other. Using velocity measurements generally leads to faster estimation of the attitude and IMU errors, as position is a further integration step from them. However, at low measurement-update rates, around 1 Hz, position measurements can perform better due to the averaging of measurement noise due to lever-arm vibration. Using time-averaged velocity measurements combines the benefits of both measurement types [2]. Faster measurement update or averaging rates also reduce state estimation biases caused by synchronization of the vibration and measurement-update cycles obstructing the averaging out of vibration-induced noise [3].

The system model and state selection for transfer alignment is essentially the same as for the INS states in INS/GNSS integration, described in Section 12.2. However, where velocity matching is used with closed-loop correction of the aligning-INS navigation solution, the position error states may be omitted, as the position error growth over a few minutes of transfer alignment will be small. Position is also reset at the reinitialization phase. The error in the assumed lever arm between the aligning INS and reference navigation system is sometimes estimated. However, it is difficult to observe, while lever-arm errors of up to 0.5m are easily tolerated.

The measurement model is analogous to that for loosely coupled INS/GNSS integration (Section 12.3.1). The measurement innovation comprises the difference between the reference navigation system and corrected aligning-INS velocity solutions, accounting for the lever arm from the reference to the aligning INS, \mathbf{l}_{rb}^b , which is assumed here to be well known. Note that r denotes the reference-navigation-system body frame. For the implementations resolved in the ECI, ECEF, and local navigation frames, the measurement innovations are, respectively,

$$\delta \mathbf{z}_{V,k}^{i-} = [\hat{\mathbf{v}}_{irR}^i - \hat{\mathbf{v}}_{ib}^i + \hat{\mathbf{C}}_b^i(\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{rb}^b)]_k \quad (13.1)$$

$$\delta \mathbf{z}_{V,k}^{e-} = [\hat{\mathbf{v}}_{erR}^e - \hat{\mathbf{v}}_{eb}^e + \hat{\mathbf{C}}_b^e(\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{rb}^b) - \boldsymbol{\Omega}_{ie}^e \hat{\mathbf{C}}_b^e \mathbf{l}_{rb}^b]_k \quad (13.2)$$

and

$$\delta \mathbf{z}_{V,k}^{n-} = [\hat{\mathbf{v}}_{erR}^n - \hat{\mathbf{v}}_{eb}^n + \hat{\mathbf{C}}_b^n(\hat{\boldsymbol{\omega}}_{ib}^b \wedge \mathbf{l}_{rb}^b) - (\hat{\boldsymbol{\Omega}}_{ie}^n + \hat{\boldsymbol{\Omega}}_{en}^n) \hat{\mathbf{C}}_b^n \mathbf{l}_{rb}^b]_k \quad (13.3)$$

where the subscript R denotes reference-navigation-system-indicated and $\boldsymbol{\Omega}_{ie}^e$, $\boldsymbol{\Omega}_{ie}^n$, and $\boldsymbol{\Omega}_{en}^n$ are given by (5.18), (5.34), and (5.37), respectively. The aligning-INS and reference velocity solutions must be time synchronized as described in Section 3.3.4.

Defining the state vector as

$$\mathbf{x}^\gamma = \begin{pmatrix} \delta\psi_{\gamma b}^\gamma \\ \delta v_{\beta b}^\beta \\ \mathbf{b}_a \\ \mathbf{b}_g \\ \vdots \end{pmatrix} \quad \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (13.4)$$

where $\delta\psi_{\gamma b}^\gamma$ is the attitude error, $\delta v_{\beta b}^\beta$ is the velocity error, \mathbf{b}_a is the accelerometer biases, and \mathbf{b}_g is the gyro biases of the aligning INS, the measurement matrix is

$$\mathbf{H}_{V,k}^\gamma = (\mathbf{H}_{v1}^\gamma \quad -\mathbf{I}_3 \quad \mathbf{0}_3 \quad \mathbf{H}_{v5}^\gamma \quad \mathbf{0})_k \quad \gamma \in i, e, n \quad (13.5)$$

where \mathbf{H}_{v1}^γ and \mathbf{H}_{v5}^γ are obtained from (12.86), (12.88), and (12.90) by substituting $-\mathbf{l}_{rb}^b$ for \mathbf{l}_{ba}^b . The coupling of the attitude errors and gyro biases into the measurements through the lever arm is weak, so a suitable approximation is

$$\mathbf{H}_{V,k}^\gamma \approx (\mathbf{0}_3 \quad -\mathbf{I}_3 \quad \mathbf{0}_3 \quad \mathbf{0}_3 \quad \mathbf{0})_k \quad \gamma \in i, e, n \quad (13.6)$$

The measurement noise arises mainly from lever-arm vibration and time-synchronization errors. It thus depends on the host aircraft and where the aligning INS is mounted. For optimum performance, a suitable value for the measurement noise covariance matrix, \mathbf{R}_V , should be determined for each scenario. However, in practice, a “worst-case” value is often assumed.

The attitude and IMU errors are determined mainly from the time evolution of the velocity error as explained in Section 12.3.4. The observability of these errors from velocity-matching measurements is the same as for INS/GNSS integration, discussed in Section 12.2.1. To observe the heading error and separate the roll and pitch errors from the horizontal accelerometer biases, the host vehicle must undergo significant maneuvering, including turns. Without this, alignment performance is significantly degraded. An s-weave alignment maneuver is typically performed [4]. Note that the heading calibration degrades slightly between the end of the maneuver and completion of transfer alignment.

13.1.2 Rapid Transfer Alignment

Rapid transfer alignment algorithms add attitude-measurement matching to the linear-measurement matching of conventional transfer alignment. This is designed to speed up the estimation of attitude errors, enabling a transfer alignment to take place within 10 seconds [5], noting that longer periods are still needed to calibrate the IMU errors. Rapid transfer alignment reduces the maneuver requirement, removing the need for turns if a wing rock (pair of rolls) is performed. It also prevents subsequent degradation of the heading alignment.

Rapid transfer alignment was first demonstrated on a helicopter [6]. Although, there are significant levels of vibration, the maneuver-dependent lever-arm flexure

is limited. However, where the aligning INS is mounted on a wing pylon of a fixed-wing aircraft, the relative orientation of the aligning INS and the reference can change significantly as the aircraft maneuvers due to wing flexure. The roll relative orientation during roll maneuvers is particularly affected and can change by a few degrees, severely biasing some of the attitude and IMU-error estimates if the Kalman filter does not model the flexure [7, 8]. This happens because the forces in the aircraft body frame are significantly different in roll maneuvers to those in level flight and coordinated turns, as Figure 13.3 illustrates. The simplest solution to this problem is to limit attitude-measurement matching to the heading component, as this is least affected by flexure, while the heading attitude error is most difficult to observe using conventional transfer alignment. Alternatively, three-component attitude matching can be made to work by estimating the coefficients of the relative orientation's variation with specific force as additional Kalman filter states [8].

The full attitude-matching measurement innovation, $\delta\mathbf{z}_{A,k}^{\gamma^-}$, is given by

$$\mathbf{I}_3 + [\delta\mathbf{z}_{A,k}^{\gamma^-} \wedge] = \hat{\mathbf{C}}_r^\gamma \hat{\mathbf{C}}_b^r \hat{\mathbf{C}}_r^b \quad \gamma \in i, e, n \quad (13.7)$$

where the estimated relative orientation, assumed to be a small angle, is given by [8]

$$\hat{\mathbf{C}}_b^r = \mathbf{I}_3 + [\hat{\boldsymbol{\psi}}_{rb} \wedge] \quad (13.8)$$

$$\hat{\boldsymbol{\psi}}_{rb} = \hat{\boldsymbol{\psi}}_{rb,s} + \begin{pmatrix} 0 & \hat{\eta}_{xy} & \hat{\eta}_{xz} \\ \hat{\eta}_{yx} & 0 & \hat{\eta}_{yz} \\ \hat{\eta}_{zx} & \hat{\eta}_{zy} & 0 \end{pmatrix} \begin{pmatrix} f_{ib,x}^b \\ f_{ib,y}^b \\ f_{ib,z}^b + g \end{pmatrix}$$

where $\boldsymbol{\psi}_{rb,s}$ is the static relative orientation, $\boldsymbol{\eta} = \{\eta_{xy}, \eta_{xz}, \eta_{yx}, \eta_{yz}, \eta_{zx}, \eta_{zy}\}$ are the flexure coefficients, and g is the acceleration due to gravity. Defining the state vector as

$$\mathbf{x}^\gamma = \begin{bmatrix} \delta\boldsymbol{\psi}_{rb}^\gamma \\ \delta\mathbf{v}_{\beta b}^\gamma \\ \mathbf{b}_a \\ \mathbf{b}_g \\ \boldsymbol{\psi}_{rb,s} \\ \boldsymbol{\eta} \\ \vdots \end{bmatrix} \quad \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (13.9)$$

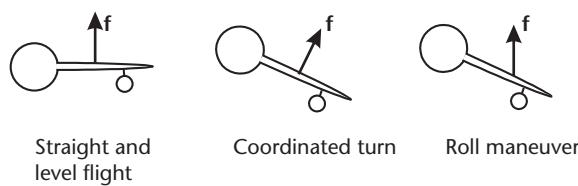


Figure 13.3 Specific forces on an aircraft wing during different maneuvers.

the attitude measurement matrix is

$$\mathbf{H}_{A,k}^{\gamma} = (-\mathbf{I}_3 \quad \mathbf{0}_3 \quad \mathbf{0}_3 \quad \mathbf{0}_3 \quad \hat{\mathbf{C}}_b^{\gamma} \quad \mathbf{H}_{a6}^{\gamma} \quad \mathbf{0})_k \quad \gamma \in i, e, n \quad (13.10)$$

where

$$\mathbf{H}_{a6}^{\gamma} = \hat{\mathbf{C}}_b^{\gamma} \begin{pmatrix} f_{ib,y}^b & f_{ib,z}^b + g & 0 & \mathbf{0}_3 & 0 & 0 \\ 0 & 0 & f_{ib,x}^b & f_{ib,z}^b + g & 0 & 0 \\ 0 & 0 & 0 & 0 & f_{ib,x}^b & f_{ib,y}^b \end{pmatrix} \quad \gamma \in i, e, n \quad (13.11)$$

With a tactical-grade IMU, the posttransfer-alignment position drift is typically within 10m per axis over the first 60 seconds, excluding errors inherited from the reference navigation system [1, 9]. A large boost immediately after the launch of the aligning INS's host vehicle can degrade this as it increases the coupling of attitude and accelerometer scale factor errors into the position and velocity solution.

13.1.3 Reference Navigation System

An integrated INS/GNSS navigation system provides a more accurate reference for transfer alignment than a stand-alone INS, as the velocity solution does not drift. It can also provide a better reference than stand-alone GNSS, as the velocity is less noisy and attitude measurements are available [10]. However, an INS/GNSS velocity can exhibit a transient when reacquisition of GNSS following a long outage leads to a large correction to the integrated navigation solution. Such a transient can disrupt transfer alignment as the Kalman filter wrongly attributes the velocity change to attitude and IMU errors in the aligning INS [8]. There are three main options for resolving this.

The simplest solution is to use a stand-alone INS as the reference. A velocity correction from the host aircraft's INS/GNSS solution is then applied alongside the position reset at the reinitialization phase at the end of transfer alignment. The degradation of the attitude and IMU-error estimates is minimal. However, if the vertical channel of the reference INS is baro-aided, the vertical velocity is subject to transients, which are difficult to model in the transfer-alignment Kalman filter. Therefore, velocity-measurement matching is limited to the horizontal components. This degrades the vertical navigation performance of the aligned INS by about a factor of two.

The optimal solution to the reference transient problem is to transmit to the aligning INS the velocity (and attitude) corrections applied to the reference INS. At the aligning INS, they are applied directly to the navigation solution, bypassing the Kalman filter. Consequently, no transients are seen in the transfer alignment measurement innovations [8]. The third approach is to implement innovation sequence monitoring, as described in Sections 15.3.2 and 15.3.3 [8].

13.2 Quasi-Stationary Alignment with Unknown Heading

Quasi-stationary alignment operates only when the INS is approximately stationary with respect to the Earth. This information is used as a reference, against which the velocity, attitude, and IMU errors are calibrated. Vibratory motion and small displacements due to human activity, such as loading, boarding, and fuelling, are treated as measurement noise.

Three types of measurements may be used: specific force, velocity, and position displacement or integrated velocity. In a low-vibration environment, specific-force measurements [11] provide faster estimation of the attitude and IMU errors, as the measurements are fewer integration steps away from these errors. However, in a high-vibration environment, position-displacement [12] or integrated velocity [13] measurements give better performance, as the standard deviation of the position displacement is correctly modeled as a constant, rather than growing with time.

Where the heading is unknown at the start of the quasi-stationary alignment, it can be determined during the alignment using indirect gyrocompassing, provided the gyros are sufficiently accurate, as discussed in Section 5.5.2. With a known pitch and roll, but initially unknown heading, the wander-azimuth coordinate frame (Section 2.1.5) must be used. The algorithm described next uses an error-state Kalman filter with position displacement measurements.

The inertial navigation equations may be simplified, as the average Earth-referenced velocity is zero. Furthermore, the north component of the acceleration due to gravity is neglected, as it is very small near the Earth's surface and the direction of the wander-azimuth frame axes with respect to north is initially unknown. The attitude may be updated using

$$\hat{\mathbf{C}}_b^w(+) \approx \hat{\mathbf{C}}_b^w(-) (\mathbf{I}_3 + \boldsymbol{\Omega}_{ib}^b) - \omega_{ie} \tau_i \begin{pmatrix} 0 & \sin L_b & -\sin \psi_{nw} \cos L_b \\ -\sin L_b & 0 & -\cos \psi_{nw} \cos L_b \\ \sin \psi_{nw} \cos L_b & \cos \psi_{nw} \cos L_b & 0 \end{pmatrix} \hat{\mathbf{C}}_b^w(-) \quad (13.12)$$

with orthogonalization and normalization as described in Section 5.4.2. The velocity and position displacement are then updated using

$$\mathbf{v}_{eb}^w(+) \approx \mathbf{v}_{eb}^w(-) + \mathbf{f}_{ib}^w \tau_i + \begin{pmatrix} 0 \\ 0 \\ g_{b,0}^n(L_b, b_b) \end{pmatrix} \tau_i \quad (13.13)$$

and

$$\Delta \mathbf{r}_{eb}^w(+) = \Delta \mathbf{r}_{eb}^w(-) + \frac{\tau_i}{2} (\mathbf{v}_{eb}^w(-) + \mathbf{v}_{eb}^w(+)) \quad (13.14)$$

The latitude, L_b , and height, h_b , obtained from a position initialization procedure (Section 5.5.1), are assumed constant. It is also assumed that the specific-force and angular-rate measurements are corrected using the IMU-error estimates and that closed-loop correction of the position displacement, velocity, attitude, and wander angle, ψ_{nw} , from the Kalman filter takes place (see Sections 3.2.6 and 12.1.1).

The sine and cosine of the wander angle are treated as separate parameters, both in the navigation equations and the Kalman filter, in order to maintain linearity in the latter. As both have zero-mean distributions, they can each be initialized at zero and then made consistent using $\cos^2 \psi + \sin^2 \psi = 1$, once confident estimates of both have been obtained.

A suitable Kalman filter state vector is thus

$$\mathbf{x}^w = \begin{bmatrix} \delta\psi_{wb}^w \\ \delta\mathbf{v}_{eb}^w \\ \delta\mathbf{r}_{eb}^w \\ \delta\sin \psi_{nw} \\ \delta\cos \psi_{nw} \\ \mathbf{b}_a \\ \mathbf{b}_g \\ \vdots \end{bmatrix} \quad (13.15)$$

where $\delta\psi_{wb}^w$, $\delta\mathbf{v}_{eb}^w$, and $\delta\mathbf{r}_{eb}^w$ are the attitude, velocity, and position errors (see Section 5.6); $\delta\sin \psi_{nw}$ and $\delta\cos \psi_{nw}$ are the errors in the sine and cosine of the wander angle; and \mathbf{b}_a and \mathbf{b}_g are the accelerometer and gyro biases. The state-propagation equations that form the system model may be simplified from those described in Section 12.2 by assuming $\mathbf{v}_{eb}^w \approx 0$, giving

$$\begin{aligned} \delta\dot{\psi}_{wb}^w &= -[\hat{\boldsymbol{\omega}}_{ie}^w \wedge] \delta\psi_{wb}^w + \omega_{ie} \cos \hat{L}_b \left(\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \delta\sin \psi_{nw} - \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \delta\cos \psi_{nw} \right) + \hat{\mathbf{C}}_b^w \mathbf{b}_g \\ \delta\dot{\mathbf{v}}_{eb}^w &\approx -[(\hat{\mathbf{C}}_b^w \hat{\mathbf{f}}_{ib}) \wedge] \delta\psi_{wb}^w + \hat{\mathbf{C}}_b^w \mathbf{b}_a \\ \delta\dot{\mathbf{r}}_{eb}^w &= \delta\mathbf{v}_{eb}^w \end{aligned} \quad (13.16)$$

where

$$\hat{\boldsymbol{\omega}}_{ie}^w = \omega_{ie} \begin{pmatrix} \cos \hat{\psi}_{nw} \cos \hat{L}_b \\ -\sin \hat{\psi}_{nw} \cos \hat{L}_b \\ \sin \hat{L}_b \end{pmatrix} \quad (13.17)$$

The wander-angle errors prevent compensation of the attitude, \mathbf{C}_b^w , for the rotation of the Earth. It is the propagation of these errors through the system

model that enables the Kalman filter to calibrate the wander angle and hence the heading, $\psi_{nb} = \psi_{nw} + \psi_{wb}$, from the position-displacement measurements (see Section 12.3.4).

The position-displacement measurements are always zero, so the measurement innovation for quasi-stationary alignment is simply

$$\delta \mathbf{z}_{Q,k}^{w-} = -\Delta \hat{\mathbf{r}}_{eb,k}^w \quad (13.18)$$

while the measurement matrix is

$$\mathbf{H}_{Q,k} = (0_3 \ 0_3 \ -\mathbf{I}_3 \ 0_{1,3} \ 0_{1,3} \ 0_3 \ 0_3 \ 0) \quad (13.19)$$

The measurement noise covariance represents the variance of the position displacement due to vibration and disturbance. Depending on the relationship between the vibration frequency and measurement-update rate, it may be necessary to treat the measurement noise as time correlated (see Section 3.4.2). Thus, \mathbf{R}_Q , is best determined empirically for each application.

It typically takes between 30 seconds and 60 seconds to determine the heading to within about 2° , enabling the small angle approximation to be made, and a few minutes to obtain the heading accuracy permitted by the gyro bias.

13.3 Quasi-Stationary Fine Alignment and Zero Velocity Updates

Where the heading is known within a few degrees, quasi-stationary fine alignment can proceed using standard inertial navigation equations (Sections 5.1–5.2) and a Kalman filter state vector and system model of the same form as INS/GNSS and multisensor integration (Chapters 12 and 14). Thus, the same Kalman filter used in the navigation phase can also accept the stationarity or zero-velocity measurements during the initial alignment phase. Furthermore, if the host vehicle or user is often stationary during the navigation phase, zero velocity updates may be used to maintain INS alignment and calibration.

ZVUs are particularly useful in poor GNSS signal environments, as often found for land vehicle navigation in urban areas [14] and for pedestrian navigation [15]. Although ZVUs do not provide absolute position information, the Kalman filter system model builds up information on the correlation between the velocity and position errors in the off-diagonal elements of the error covariance matrix, \mathbf{P} . This enables a ZVU to correct most of the position drift since the last measurement update, ZVU or otherwise [15].

For pedestrian navigation with a shoe-mounted IMU, a ZVU may be performed on every step. This enables inertial navigation with a low position-drift rate to be performed with very-low-cost automotive-grade inertial sensors [16–18].

For quasi-stationary fine alignment operating over a few minutes, position-displacement measurements enable the best modeling of the vibration. The measurement innovation is thus

$$\delta \mathbf{z}_{Q,k}^{n-} = \hat{\mathbf{p}}_b(t_0) - \hat{\mathbf{p}}_b(t) \quad (13.20)$$

in the local navigation frame,

$$\delta \mathbf{z}_{Q,k}^{e-} = \hat{\mathbf{r}}_{eb}^e(t_0) - \hat{\mathbf{r}}_{eb}^e(t) \quad (13.21)$$

in the ECEF frame, and

$$\delta \mathbf{z}_{Q,k}^{i-} = \begin{pmatrix} \cos \omega_{ie}(t - t_0) & -\sin \omega_{ie}(t - t_0) & 0 \\ \sin \omega_{ie}(t - t_0) & \cos \omega_{ie}(t - t_0) & 0 \\ 0 & 0 & 1 \end{pmatrix} \hat{\mathbf{r}}_{ib}^i(t_0) - \hat{\mathbf{r}}_{ib}^i(t) \quad (13.22)$$

in the ECI frame, where t_0 is the alignment start time. If the state vector is

$$\mathbf{x}^n = \begin{bmatrix} \delta \boldsymbol{\psi}_{nb}^n \\ \delta \mathbf{v}_{eb}^n \\ \delta \mathbf{p}_b \\ \mathbf{b}_a \\ \mathbf{b}_a \\ \vdots \end{bmatrix} \quad \mathbf{x}^\gamma = \begin{bmatrix} \delta \boldsymbol{\psi}_{\gamma b}^\gamma \\ \delta \mathbf{v}_{\gamma b}^\gamma \\ \delta \mathbf{r}_{\gamma b}^\gamma \\ \mathbf{b}_a \\ \mathbf{b}_a \\ \vdots \end{bmatrix} \quad \gamma \in i, e \quad (13.23)$$

where the curvilinear position error, $\delta \mathbf{p}_b$, is defined by (12.50), the measurement matrix is

$$\mathbf{H}_{Q,k} = (0_3 \ 0_3 \ -\mathbf{I}_3 \ 0_3 \ 0_3 \ 0) \quad (13.24)$$

For ZVUs, where the INS may only be stationary briefly, velocity measurements are generally better. The measurement innovation is thus

$$\delta \mathbf{z}_{Z,k}^{\gamma-} = -\hat{\mathbf{v}}_{eb,\gamma}^{\gamma} \quad \gamma \in e, n \quad (13.25)$$

or

$$\delta \mathbf{z}_{Z,k}^{i-} = \boldsymbol{\Omega}_{ie}^i \hat{\mathbf{r}}_{ib,k}^i - \hat{\mathbf{v}}_{ib,k}^i \quad (13.26)$$

and the measurement matrix is

$$\mathbf{H}_{Z,k}^\gamma = (0_3 \ -\mathbf{I}_3 \ 0_3 \ 0_3 \ 0_3 \ 0) \quad \gamma \in e, n \quad (13.27)$$

or

$$\mathbf{H}_{Z,k}^n = (0_3 \ -\mathbf{I}_3 \ \boldsymbol{\Omega}_{ie}^i \ 0_3 \ 0_3 \ 0) \quad (13.28)$$

To perform ZVUs, the navigation system must detect when it is stationary. For pedestrian navigation, an acceleration threshold, applied to

$||\mathbf{f}_{ib}^b| - g(L_b, b_b)|$ over a time window of about 0.5 seconds, is suitable [15], while for land vehicle navigation, a velocity threshold, normalized with the velocity uncertainty, may be applied [14].

For land vehicles, the velocity along the rear-wheel axis may be assumed to be zero, an example of a nonholonomic constraint [19]. This enables single-dimensional partial ZVUs to be continuously applied, constraining the cross-track position error growth. The lever arm between the axle and IMU-body frames should be accounted for (see Section 10.3).

References

- [1] Groves, P. D., “Optimising the Transfer Alignment of Weapon INS,” *Journal of Navigation*, Vol. 56, No. 3, 2003, pp. 323–335.
- [2] Spalding, K., “An Efficient Rapid Transfer Alignment Filter,” *Proc. AIAA Guidance, Navigation and Control Conference*, Hilton Head Island, SC, August 1992, pp. 1276–1286.
- [3] Wendel, J., and G. Trommer, “Impact of Mechanical Vibrations on the Performance of Integrated Navigation Systems and an Optimal IMU Specification,” *Proc. ION 57th AM*, Albuquerque, NM, June 2001, pp. 614–621.
- [4] Titterton, D. H., and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed., Stevenage, U.K.: IEE, 2004.
- [5] Kain, J. E., and J. R. Cloutier, “Rapid Transfer Alignment for Tactical Weapon Applications,” *Proc. AIAA Guidance, Navigation and Control Conference*, Boston, MA, August 1989, pp. 1290–1300.
- [6] Graham, W., and K. Shortelle, “Advanced Transfer Alignment for Inertial Navigators (A-Train),” *Proc. ION NTM*, Anaheim, CA, January 1995, pp. 113–124.
- [7] Rogers, R. M., *Applied Mathematics in Integrated Navigation Systems*, Reston, VA: AIAA, 2000.
- [8] Groves, P. D., G. G. Wilson, and C. J. Mather, “Robust Rapid Transfer Alignment with an INS/GPS Reference,” *Proc. ION NTM*, San Diego, CA, January 2002, pp. 301–311.
- [9] Graham, W. R., K. J. Shortelle, and C. Rabourn, “Rapid Alignment Prototype (RAP) Flight Test Demonstration,” *Proc. ION NTM*, Long Beach, CA, January 1998, pp. 557–568.
- [10] Groves, P. D., C. A. Littlefield, and D. C. Long, “The Need for Transfer Alignment in a GPS Jamming Environment and Optimization for MEMS IMU,” *Proc. ION GNSS 2004*, Long Beach, CA, September 2004, pp. 775–783.
- [11] Farrell, J. A., and M. Barth, *The Global Positioning System and Inertial Navigation*, New York: McGraw-Hill, 1999.
- [12] Savage, P. G., *Strapdown Analytics, Parts 1 and 2*, Maple Plain, MN: Strapdown Associates, 2000.
- [13] Hua, C., “Gyrocompass Alignment with Base Motions: Result for a 1nmi/h INS/GPS System,” *Navigation: JION*, Vol. 47, No. 2, 2000, pp. 65–74.
- [14] Grejner-Brzezinska, D. A., Y. Yi, and C. K. Toth, “Bridging Gaps in Urban Canyons: The Benefits of ZUPTs,” *Navigation: JION*, Vol. 48, No. 4, 2001, pp. 217–225.
- [15] Mather, C. J., P. D. Groves, and M. R. Carter, “A Man Motion Navigation System Using High Sensitivity GPS, MEMS IMU and Auxiliary Sensors,” *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 2704–2714.
- [16] Brand, T. J., and R. E. Phillips, “Foot-to-Foot Range Measurements As an Aid to Personal Navigation,” *Proc. ION 59th AM*, Albuquerque, NM, June 2003, pp. 113–125.
- [17] Foxlin, E., “Pedestrian Tracking with Shoe-Mounted Inertial Sensors,” *IEEE Computer Graphics and Applications Magazine*, November/December 2005, pp. 38–46.

- [18] Godha, S., G. Lachapelle, and M. E. Cannon, "Integrated GPS/INS System for Pedestrian Navigation in a Signal Degraded Environment," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 2151–2164.
- [19] El-Sheimy, N., and X. Niu, "The Promise of MEMS to the Navigation Community," *Inside GNSS*, March/April 2007, pp. 46–56.

Selected Bibliography

Wendel, J., *Integrierte Navigationssysteme: Sensorfusion, GPS und Inertiale Navigation*, München, Deutschland: Oldenbourg Verlag, 2007.

Multisensor Integrated Navigation

This chapter describes how terrestrial radio navigation, dead reckoning, and feature-matching navigation systems may be integrated with INS, GNSS, and each other. It follows on from the description of INS/GNSS integration in Chapter 12.

Different combinations of navigation sensors are suited to different applications, depending on the environment, dynamics, budget, accuracy requirements, and the degree of robustness or integrity required.

For commercial airliners and most military aircraft, INS and GNSS form the core of the navigation system, with further sensors, such as baro, magnetic compass, Loran, DME/TACAN, and, for military applications, TRN, providing enhanced robustness. ILS is commonly used for landing, but tends not to be integrated with the other navigation sensors. Helicopters tend to use Doppler radar instead of or as well as inertial navigation, while many light aircraft and UAVs use an AHRS rather than an INS.

Ships also use INS and GNSS for navigation with respect to the Earth (as opposed to the water), with a magnetic compass maintaining the heading calibration. Loran may be used for extra robustness, where available, while Doppler sonar provides added robustness in coastal areas and inland waterways. Submarines and UAVs can only use GNSS and other radio navigation systems when they surface. Underwater, they rely on inertial navigation, various sonar devices, a depth sensor, and sometimes gravity gradiometry.

Road vehicles typically combine GNSS with odometers, magnetic compass, baro, and map-matching algorithms, while trains may also use Doppler radar. For pedestrian navigation, GNSS may be combined with a variety of sensors. Cell phone, UWB, and WLAN positioning may supplement GNSS indoors and in urban areas. A low-cost shoe-mounted IMU may be used for inertial navigation with frequent ZVUs, while body-mounted inertial sensors may be used for PDR, in both cases augmented with a baro and magnetic compass. Loran may also be used for land applications.

Spacecraft position and velocity is principally determined using force models with occasional position fixes. This is more accurate than inertial navigation except during periods of significant maneuvering. Position fixes in low Earth orbit can be obtained from GNSS. Otherwise, combinations of tracking from Earth, two-way radio ranging to ground stations, and visual matching of planetary features are used. The spacecraft attitude solution is maintained using a gyro triad, kept aligned with attitude fixes from a star tracker, though multiantenna GNSS attitude may also be used in low Earth orbit.

Section 14.1 describes and compares the different architectures that may be used to integrate measurements from three or more different navigation systems. Sections 14.2, 14.3, and 14.4 then discuss the integration issues and describe system and measurement models for terrestrial radio navigation, dead reckoning, and feature matching, respectively. Integration of INS and GNSS into multisensor integration architectures is essentially the same as for INS/GNSS integration, described in Chapter 12.

14.1 Integration Architectures

There are many ways of combining information from multiple navigation systems. The design of the integration architecture is a tradeoff between maximizing the accuracy and robustness of the navigation solution, minimizing the complexity, and optimizing the processing efficiency. It must also account for the characteristics of the different navigation technologies. Inertial and dead-reckoning position solutions drift over time, but may be calibrated using positioning systems. GNSS and terrestrial radio navigation systems require a minimum number of signals to form a navigation solution. Feature-matching systems require an approximate position solution to determine which region of their database to search; some, such as TRN, also require a velocity solution.

Many navigation sensors exhibit biases and other systematic errors, which can be calibrated in an integrated navigation system. However, such calibration can result in faults in one navigation sensor contaminating the calibration of another. Fault detection and integrity monitoring is described in Chapter 15 and can be implemented for all multisensor integration architectures. However, integrity monitoring is more processor intensive for some integration architectures than for others.

The design of an integrated navigation system can be severely constrained by the need to combine equipment from different manufacturers. Where raw sensor or ranging measurements are not available, the systems integrator may have to work with a “black box” navigation solution with no information about its error characteristics, such as covariances, correlation times, or even uncertainty in many cases. Such systems are also limited in terms of what feedback information they may accept.

This section describes the different integration architectures and discusses their benefits and drawbacks. Simple least-squares integration is described first, followed by the cascaded, centralized, and federated architectures, and then a discussion of hybrid architectures. The section concludes with discussions of the total-state and error-state Kalman filters, including prediction and timing.

14.1.1 Least-Squares Integration

Least-squares integration, shown in Figure 14.1, is the simplest way of combining information from different navigation systems. Each system, denoted by index i , provides a position or position and velocity solution, $\hat{\mathbf{x}}_i$, and an associated error covariance matrix, \mathbf{P}_{ii} . These are combined with a snapshot, or single-point, fusing

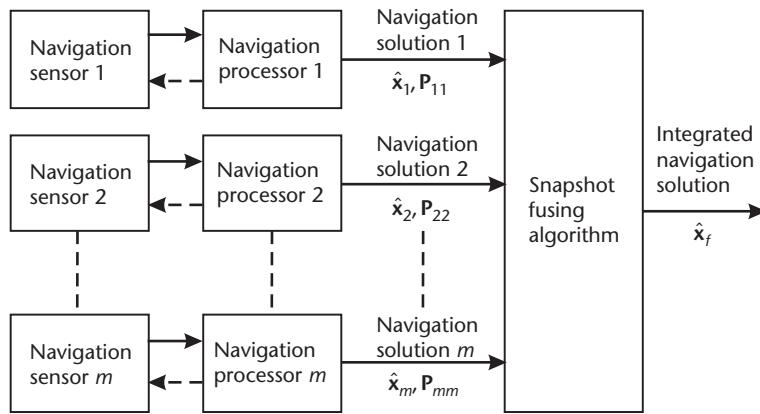


Figure 14.1 Least-squares integration architecture.

algorithm, analogous to the single-point GNSS navigation solution described in Section 7.5.1.

The integrated navigation solution, $\hat{\mathbf{x}}_f$, is simply the weighted least-squares combination of the m individual solutions, given by

$$\hat{\mathbf{x}}_f = (\mathbf{H}^T \mathbf{C}^{+ -1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{C}^{+ -1} \begin{pmatrix} \hat{\mathbf{x}}_1 \\ \hat{\mathbf{x}}_2 \\ \vdots \\ \hat{\mathbf{x}}_m \end{pmatrix} \quad (14.1)$$

where the measurement matrix, \mathbf{H} , is a column of $m n \times n$ identity matrices, where n is the number of components of \mathbf{x} and \mathbf{C}^+ is the covariance of the measurement residuals, given by

$$\mathbf{C}^+ = \begin{pmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} & \dots & \mathbf{P}_{1m} \\ \mathbf{P}_{21} & \mathbf{P}_{22} & \dots & \mathbf{P}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m1} & \mathbf{P}_{m2} & \dots & \mathbf{P}_{mm} \end{pmatrix} \quad (14.2)$$

The covariance of the fused navigation solution is

$$\mathbf{P}_{ff} = (\mathbf{H}^T \mathbf{C}^{+ -1} \mathbf{H})^{-1} \quad (14.3)$$

Where each navigation solution uses different information to obtain its navigation solution, the errors of the different navigation solutions will be uncorrelated, so $\mathbf{P}_{ij} = 0$ for $i \neq j$. This simplifies (14.1) and (14.3) to

$$\hat{\mathbf{x}}_f = \mathbf{P}_{ff} \sum_{i=1}^m \mathbf{P}_{ii}^{-1} \hat{\mathbf{x}}_i \quad (14.4)$$

$$\mathbf{P}_{ff} = \left(\sum_{i=1}^m \mathbf{P}_{ii}^{-1} \right)^{-1} \quad (14.5)$$

The least-squares integration architecture is suited to black box navigation systems as the fusion algorithm requires no knowledge of how the navigation system errors vary with time and there is no feedback. However, to optimally combine the different navigation solutions, accurate error covariance information is needed. Figure 14.2 shows that neglecting the off-diagonal elements of \mathbf{P} causes the accuracy to be overestimated in one direction and underestimated in another. Using the off-diagonal elements also allows incomplete navigation solutions to be fused. However, this requires the navigation systems to output the information matrix, \mathbf{P}^{-1} , instead of the error covariance matrix, as an incomplete navigation solution has infinite uncertainty in one or two directions.

Where a navigation system outputs no uncertainty information, its error covariance must be estimated by the fusing algorithm. This can be a problem for radio navigation systems, such as GNSS, where the accuracy varies with the number of satellites tracked, their geometry, and signal-to-noise ratios.

Least-squares integration has the advantages of simplicity and a low processor load. As the subsystems are completely independent, it also facilitates integrity monitoring, allowing measurement consistency checks to be used (see Section 15.4.1). However, it has fundamental limitations. It is unsuited to integration of inertial navigation or dead-reckoning systems, as it offers no means of calibrating the position drift. Instead, as an inertial or DR position degrades, it is simply weighted out of the integrated navigation solution. Least-squares integration also offers no means of combining navigation data with different times of validity, so is unsuited to fast-moving vehicles.

14.1.2 Cascaded Integration

Figure 14.3 shows a total-state cascaded integration architecture. This is similar to the least-squares architecture, but with the snapshot fusing algorithm replaced by a Kalman filter (Chapter 3). This estimates the navigation solution and can also

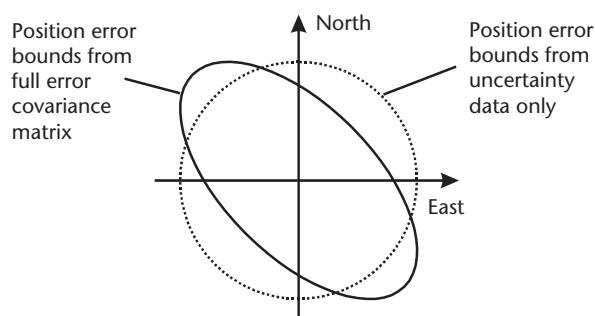


Figure 14.2 Effect of off-diagonal error covariance on position error bounds.

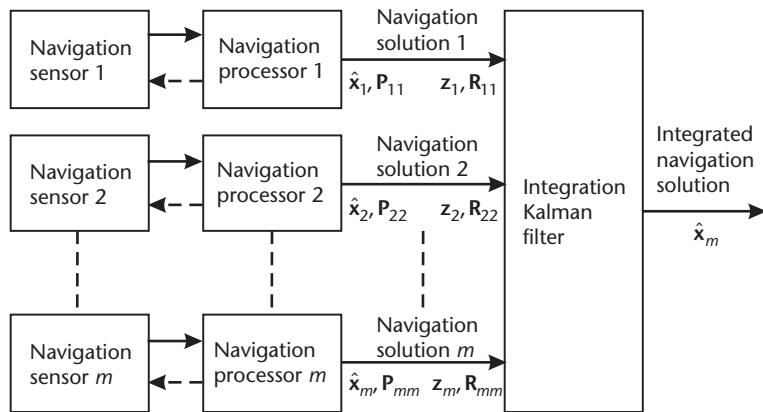


Figure 14.3 Total-state cascaded integration architecture.

estimate errors in some or all of the navigation subsystems. As a Kalman filter–based integration algorithm retains past information from the constituent navigation systems, it can maintain calibration of INS/DR position solution drift and other errors. It can also handle different sensors providing measurements at different times of validity (e.g., by using its velocity estimate to predict forward position information from one subsystem to the time of validity of another subsystem’s position measurement).

Figure 14.4 shows an error-state cascaded integration architecture. The integrated navigation solution is that of an INS or DR reference system, corrected using estimates of its position, velocity, and attitude error made by the Kalman filter integration algorithm. Open- and closed-loop correction of the reference system is discussed in Sections 12.1.1 and 14.1.7. This brings the advantage that the integrated navigation solution may be updated at a faster rate than the Kalman filter is iterated at, reducing the processor load. An example of cascaded integration

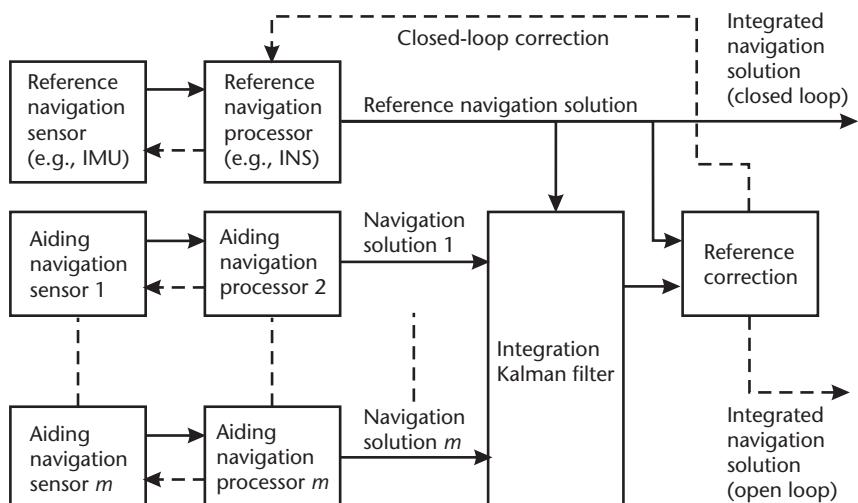


Figure 14.4 Error-state cascaded integration architecture.

is loosely coupled INS/GNSS integration, described in Section 12.1.2, where a GNSS navigation filter is used. Where an error-state Kalman filter integrates both INS and dead reckoning, one system, usually the INS, is integrated as the reference and the other as an aiding system.

Although not shown in the figures, corrections from the integration algorithm may be fed back to any of the navigation processors if their software accepts them. The reference or integrated navigation solution may also be fed back (e.g., to aid GNSS signal tracking or determine a feature matching system's search area). However, feedback reduces the independence between navigation systems, making it more difficult to detect faults by comparing navigation solutions.

In the total-state implementation, the navigation solution of each of the navigation systems forms the Kalman filter measurement vector, \mathbf{z} . In the error-state implementation, the measurement vector comprises the difference between the aiding and reference system navigation solutions. Section 12.3.1 shows how this is implemented for GNSS measurements integrated with INS. The implementation for other navigation systems is similar. Note that any data lags must be compensated, as discussed in Section 3.3.4. Measurements from different navigation systems do not have to be processed simultaneously.

A fundamental assumption of Kalman filtering is that the navigation-system errors comprise a mixture of systematic errors, estimated as states, and white noise, modeled as system noise for a reference system and measurement noise otherwise. However, the use of Kalman filters and smoothing filters in the individual navigation processors introduces time-correlated noise. The integration Kalman filter must account for this, as discussed in Section 3.4.2, to prevent instability. The simplest method is to increase the assumed measurement noise covariance, \mathbf{R} . Thus, where black box navigation systems are integrated in a cascaded architecture, their error characteristics must be determined across all operational conditions to ensure that the integration Kalman filter is correctly tuned. If this is not practical, a cascaded integration architecture should not be used. It is also difficult to handle incomplete subsystem navigation solutions in a cascaded architecture.

14.1.3 Centralized Integration

Figures 14.5 and 14.6, respectively, show the total-state and error-state implementations of the centralized integration architecture. The total-state filter is suited to integrating positioning systems only, whereas an error-state filter is suitable where INS or dead reckoning is used. In contrast to the cascaded architecture, sensor measurements rather than navigation solutions are generally input to the integration Kalman filter. Radio navigation systems provide ranging measurements; thus, tightly coupled INS/GNSS integration (Section 12.1.3) is an example of centralized integration. This enables a navigation system to contribute to the integrated navigation solution when there are insufficient signals to form its own solution. Either IMU or INS measurements are acceptable in a centralized architecture, as the inertial navigation equations do not incorporate any smoothing or estimation algorithm. DR and feature-matching systems may also output either the sensor or navigation measurements, provided they do not pass through smoothing or estimation algorithms.

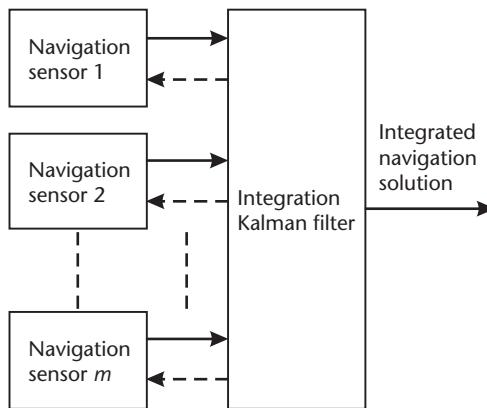


Figure 14.5 Total-state centralized integration architecture.

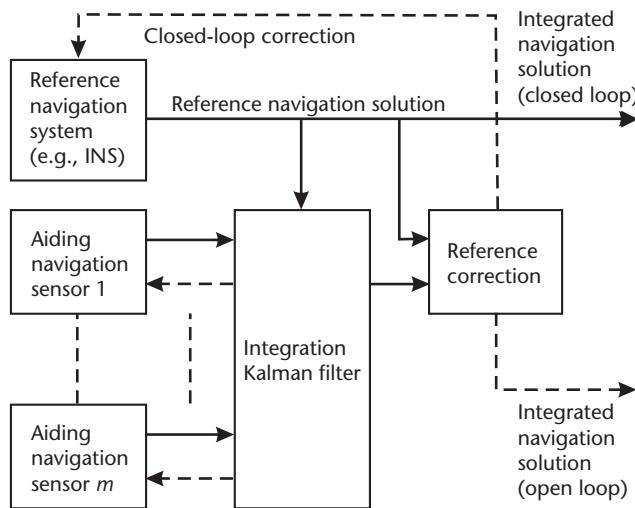


Figure 14.6 Error-state centralized integration architecture.

Processing of GNSS measurements is described in Section 7.5.2 for a total-state Kalman filter and in Section 12.3.2 for an error-state Kalman filter with an INS reference. Processing of terrestrial-radio-navigation, dead-reckoning, and feature-matching measurements is described in Sections 14.2, 14.3, and 14.4, respectively. Correct handling of data lags (see Section 3.3.4) is particularly important in centralized integration.

In a centralized integration architecture, the systematic errors and noise sources of all of the navigation sensors are modeled in the same Kalman filter. This ensures that all error correlations are accounted for, all measurements optimally weighted, and the maximum information used to calibrate each error. Furthermore, the elimination of Kalman filter cascades enables higher gains to be used before there is a stability risk. Thus, the centralized integration architecture provides the optimal navigation solution in terms of accuracy and robustness. However, this is contingent

on having the necessary information to model all sensors correctly, requiring careful design.

With all of the error sources modeled in one place, the principal disadvantage of centralized integration is a high processor load. With no independent subsystem navigation solutions available, processor-intensive parallel filters (see Section 15.4.2) are needed for applications with demanding integrity requirements. As centralized integration requires raw sensor measurements and information about their error characteristics, it is not compatible with black box navigation systems.

14.1.4 Federated Integration

In a federated-filters integration architecture, a reference inertial or dead-reckoning navigation solution is integrated separately, with each of the aiding navigation systems in a bank of local Kalman filters. Each local filter's integration with its navigation sensors may be centralized or cascaded. There are a number of different ways in which the local filter outputs may be combined to produce an integrated navigation solution. The no-reset, fusion-reset, zero-reset, and cascaded versions of federated integration are described next.

14.1.4.1 No Reset

Figure 14.7 shows the federated no-reset (FNR) integration architecture, in which the navigation solutions and reference-system error estimates from the local filters are combined with a snapshot fusing algorithm using (14.4) and (14.5) [1]. Changes in the raw reference navigation solution can be used to propagate the integrated navigation solution between fusing algorithm updates.

The FNR architecture is useful for integrating black box navigation systems that accept a common INS-aiding input without the tuning difficulties inherent in

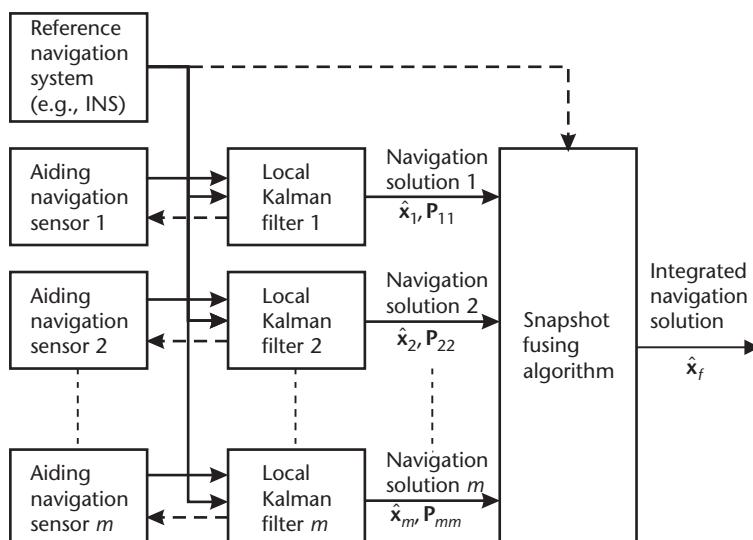


Figure 14.7 Federated no-reset integration architecture.

the cascaded approach. The local filter outputs are also suited to integrity monitoring by measurement consistency checks (Section 15.4.1). Closed-loop corrections can only be fed back to the reference system if they are accepted by all of the local filters; otherwise, local filter estimation would be disrupted by unmodeled transients. However, useful open-loop integration of an INS requires high-quality sensors.

A problem with the FNR architecture is that, as the reference navigation system is common to all of the local filters, their navigation solutions are no longer independent, so $P_{ij} \neq 0$. Consequently, the weighting of the local filter solutions is suboptimal and the error covariance of the integrated navigation solution, P_{ff} , is overoptimistic. A work-around solution to this is to overestimate the initial error covariance and system noise covariance of the reference-system states in the local filters by a factor of m , the number of local filters [2], though this does cause the Kalman gains for these states to be overestimated.

14.1.4.2 Fusion Reset

The federated fusion-reset (FFR) architecture, shown in Figure 14.8, feeds back the state estimates and error covariance from the least-squares fusing algorithm to the local filters, where they replace the corresponding states and error covariance matrix elements, the latter scaled up by a factor of m [3]. This allows calibration information to be shared between all navigation sensors at a lower processing load than centralized integration without cascading Kalman filters. The local filters may be implemented using parallel processors. However, the problem then arises of how to model the correlations between the common and local states in each local filter; one solution makes use of a conventional Kalman filter in parallel with each local filter [4].

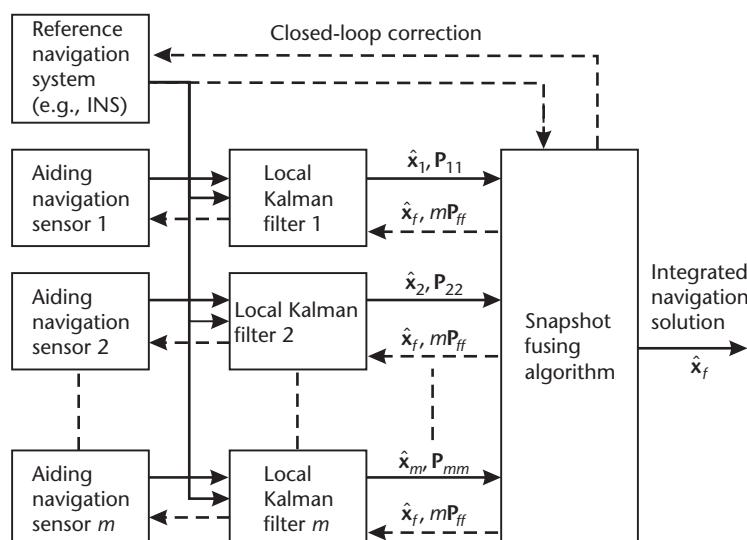


Figure 14.8 Federated fusion-reset architecture.

The FFR architecture shares many of the drawbacks of the centralized architecture: it is incompatible with black box navigation systems and does not provide independent subsystem solutions for integrity monitoring. It is also more complex than the centralized architecture and gives poorer performance [5].

14.1.4.3 Zero Reset

The federated zero-reset (FZR) architecture, shown in Figure 14.9, uses a Kalman filter to integrate the outputs of the local filters. However, Kalman filter cascading is avoided by zeroing all local filter states after they are input to the master Kalman filter, with the corresponding elements of the error covariance matrix set to their initialization values [1]. State estimates that are not passed to the master filter may be retained in the local filters. This zero reset prevents the same data from being input to the master filter more than once. Unlike in the FNR and FFR architectures, measurements from different local filters may be processed by the master filter at different times.

The FZR architecture can be useful where there is a need to process measurements at a faster rate than it is practical to run the integration algorithm. It has found use in coherent deep INS/GPS integration (Section 12.3.3), with the local filters performing the signal tracking and the master filter the integration.

14.1.4.4 Cascaded

Figure 14.10 shows a federated filters architecture with a Kalman filter integrating the outputs of the local filters without any resets. The integration of the local filters with the master filter is thus cascaded, bringing the problem of time-correlated measurement noise (Section 3.4.2). If any of the local filters are cascaded, this introduces a double cascade. Furthermore, there is error correlation between each of

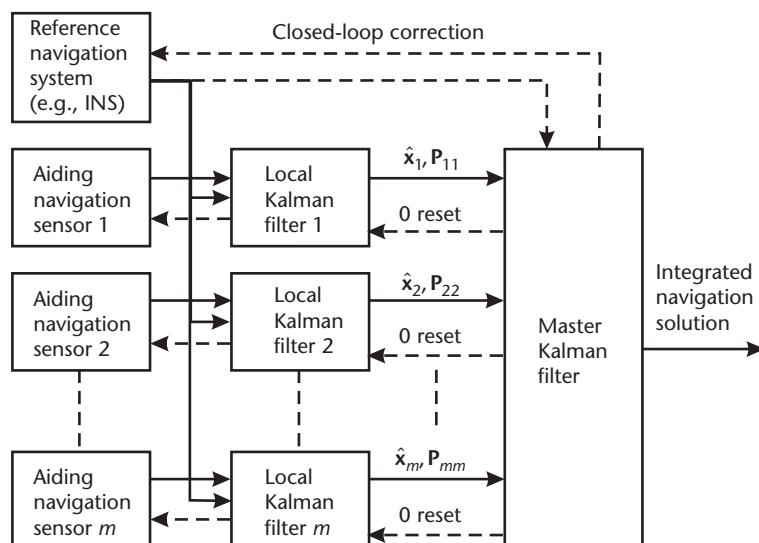


Figure 14.9 Federated zero-reset architecture.

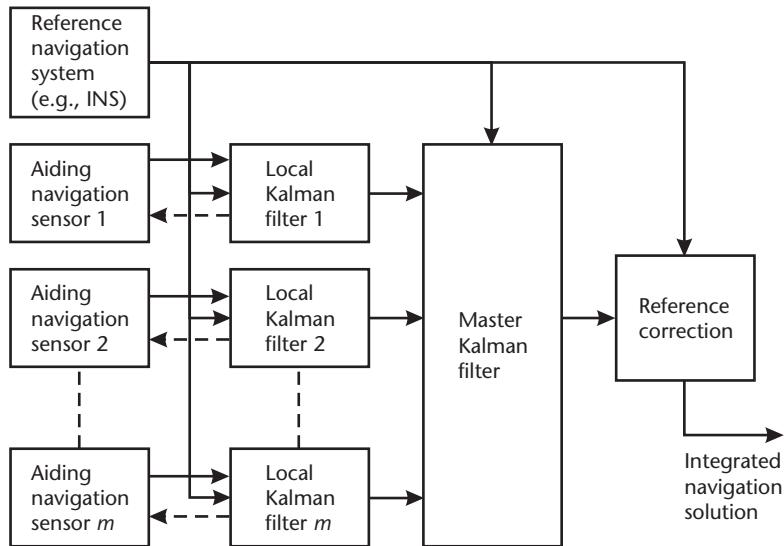


Figure 14.10 Federated filters with cascaded integration architecture.

the local filter outputs and between the local filter outputs and reference navigation system. This architecture should thus be approached with great caution, as the master Kalman filter must be tuned very carefully to avoid instability and produce realistic uncertainties. Its only advantage is compatibility with black box local filters.

14.1.5 Hybrid Integration Architectures

Different architectures may be used for different sensors in the same integrated navigation system, provided the final stage of the processing chain is common. Thus, the least-squares, FNR, and FFR architectures can be mixed, as can the centralized, cascaded, FZR, and federated-cascaded architectures. However, architectures using a snapshot least-squares fusing algorithm to produce the integrated navigation solution cannot be mixed with architectures using a Kalman filter. Hybrid architectures are typically used where constraints in the design of the constituent navigation systems prevent use of the desired architecture in all cases. Figure 14.11 depicts an example integration of INS with centralized GNSS, cascaded Loran, and federated-cascaded TRN.

14.1.6 Total-State Kalman Filter Employing Prediction

Total-state integration employing prediction is applicable to centralized and local Kalman filters integrating only positioning systems. It is also suitable for use with dead-reckoning sensors where the velocity measurement noise exceeds the variation in the host vehicle's velocity between measurements.

The state vector comprises a navigation solution, $\mathbf{x}_{\text{Nav}}^{\gamma}$, together with error states for each of the navigation systems integrated, $\mathbf{x}_{\text{Sensor}-i}$. Thus, where n sensors are combined,

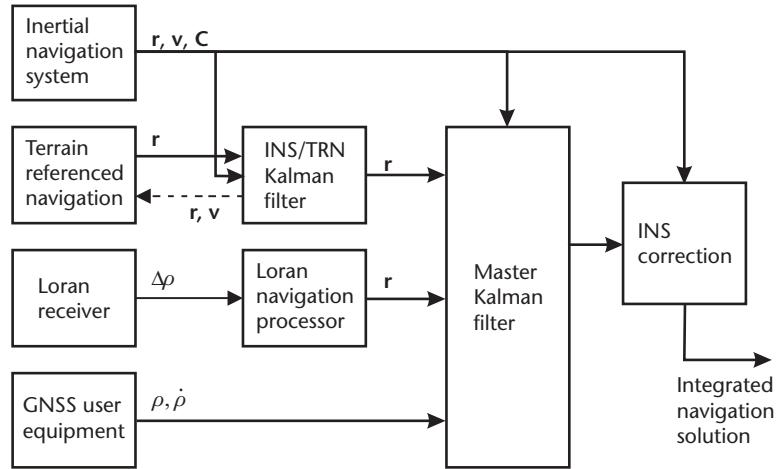


Figure 14.11 Hybrid integration of INS with centralized GNSS, cascaded Loran, and federated-cascaded TRN.

$$\mathbf{x}^\gamma = \begin{pmatrix} \mathbf{x}_{Nav}^\gamma \\ \mathbf{x}_{Sensor-1} \\ \mathbf{x}_{Sensor-2} \\ \vdots \\ \mathbf{x}_{Sensor-n} \end{pmatrix} \quad \gamma \in i, e, n \quad (14.6)$$

The ECI, ECEF, and local-navigation-frame navigation solutions are

$$\mathbf{x}_{Nav}^i = \begin{pmatrix} \mathbf{r}_{ib}^i \\ \mathbf{v}_{ib}^i \end{pmatrix}, \mathbf{x}_{Nav}^e = \begin{pmatrix} \mathbf{r}_{eb}^e \\ \mathbf{v}_{eb}^e \end{pmatrix}, \mathbf{x}_{Nav}^n = \begin{pmatrix} L_b \\ \lambda_b \\ h_b \\ \mathbf{v}_{eb}^n \end{pmatrix} \quad (14.7)$$

Where DR sensors are integrated, attitude states are added. Alternatively, total-state position and velocity integration may be combined with error-state attitude integration.

In total-state integration, the position solution is predicted forward in time using the velocity solution. This is useful for smoothing noise in navigation system measurements, aiding time synchronization, and bridging gaps in navigation system measurements (e.g., in tunnels). The accuracy of the predicted solution depends on the host-vehicle dynamics, so its uncertainty must be correctly modeled to ensure that the predictions and the sensor measurements are correctly weighted in the navigation solution. Prediction takes place in the Kalman filter's system model (Section 3.2.4). This is described in Section 7.5.2.2 for a GNSS navigation filter and is also applicable generally.

For high-dynamics applications, navigation solution prediction may be enhanced by adding acceleration states to the Kalman filter. Further improvements

can sometimes be made by adding force modeling. For most air, land, and marine applications, the forces are too complex for this to be practical. An exception is some ballistic missiles and guided shells. However, for space applications, force modeling can be very accurate. With a good model, satellite positions can be predicted to within a few meters over a complete orbit without new measurements [6].

The system models for GNSS, terrestrial radio navigation, DR, and feature-matching systems are described in Sections 7.5.2.2, 14.2, 14.3, and 14.4, respectively. Most sensor errors should be modeled as either a random walk or a first-order Markov process. For a random walk state, x_{ri} , the state dynamics and system noise variance are

$$\frac{\partial x_{ri}}{\partial t} = 0, \quad Q_{ri} = n_{ri}^2 \tau_s \quad (14.8)$$

where τ_s is the state-propagation interval, and n_{ri}^2 is the PSD of the noise producing the random walk. The state dynamics and system noise variance of an exponentially correlated fixed-variance first-order Markov process are given by (3.3) and (12.70). Error states are not necessarily estimated for all constituent navigation systems.

The Kalman filter measurement model is described in Section 3.3.5. Where navigation systems output measurements with the same time of validity, they may be processed by the Kalman filter together. Otherwise, they must be processed separately. By analogy with (12.73) to (12.77), the measurement innovation vector for each sensor is of the form

$$\delta z_i^- = \tilde{m}_i - \delta \hat{m}_i - \hat{m}_{Nav} \quad (14.9)$$

where \tilde{m}_i is the vector of measurements output by sensor i (e.g., a position solution or a set of range measurements), $\delta \hat{m}_i$ is the Kalman-filter-estimated error in those measurements, calculated from $\hat{x}_{Sensor-i}$, and \hat{m}_{Nav} is the value of m estimated from the predicted navigation solution, \hat{x}_{Nav}^γ . Note that the system model must be used to predict forward the estimated navigation solution to the measurement time of validity prior to the measurement update. The measurement matrix is then

$$H_i^\gamma = \begin{pmatrix} -\frac{\partial m}{\partial x_{Nav}^\gamma} & 0 & \frac{\partial m}{\partial x_{Sensor-i}} & 0 \end{pmatrix}_{x^\gamma = \hat{x}^\gamma} \quad (14.10)$$

noting that $\partial m / \partial x_{Sensor-j}$ is a null matrix for $j \neq i$. Measurement models for GNSS are described in Sections 7.5.2.3 and 12.3, while those for terrestrial radio navigation, DR, and feature matching are discussed in Sections 14.2, 14.3, and 14.4, respectively.

A problem that can arise in multisensor navigation systems is that, due to variation in the duty cycles and processing delays between different sensors, measurements may not become available in order of time of validity. The simplest solution is to delay processing of those measurements with the shorter lags so that measurements may be processed in time-of-validity order. However, this will delay

the integrated navigation solution. Another option is to propagate the older measurements forward through the system model to the current time of validity.

14.1.7 Error-State Kalman Filter

In error-state multisensor integration, the integrated navigation solution comprises the navigation solution of a reference navigation system, corrected using measurements from the other constituent navigation systems. The reference system must be an INS or other dead-reckoning system. Where noninertial DR is used, a sensor measuring velocity or distance traveled must be combined with attitude measurement sensors so that a navigation solution resolved about the ECI, ECEF, or local navigation frame may be maintained. The reference solution takes the place of the predicted solution in a total-state Kalman filter.

The state vector comprises error states for the reference navigation system, $\mathbf{x}_{\text{Ref}}^\gamma$, and for each of the aiding systems, $\mathbf{x}_{\text{Sensor-}i}$. With n aiding sensors, it is

$$\mathbf{x}^\gamma = \begin{pmatrix} \mathbf{x}_{\text{Re } f}^\gamma \\ \mathbf{x}_{\text{Sensor-1}} \\ \mathbf{x}_{\text{Sensor-2}} \\ \vdots \\ \mathbf{x}_{\text{Sensor-}n} \end{pmatrix} \quad \gamma \in i, e, n \quad (14.11)$$

The reference navigation system error states comprise attitude, velocity, and position errors, defined in Section 5.6, together with reference sensor errors $\mathbf{x}_{\text{Sensor-Re } f}$. Thus, for the ECI, ECEF, and local-navigation-frame error-state implementations,

$$\mathbf{x}_{\text{Re } f}^i = \begin{pmatrix} \delta\psi_{ib}^i \\ \delta\mathbf{v}_{ib}^i \\ \delta\mathbf{r}_{ib}^i \\ \mathbf{x}_{\text{Sensor-Re } f} \end{pmatrix}, \quad \mathbf{x}_{\text{Re } f}^e = \begin{pmatrix} \delta\psi_{eb}^e \\ \delta\mathbf{v}_{eb}^e \\ \delta\mathbf{r}_{eb}^e \\ \mathbf{x}_{\text{Sensor-Re } f} \end{pmatrix}, \quad \mathbf{x}_{\text{Re } f}^n = \begin{pmatrix} \delta\psi_{nb}^n \\ \delta\mathbf{v}_{nb}^n \\ \delta\mathbf{p}_b \\ \mathbf{x}_{\text{Sensor-Re } f} \end{pmatrix} \quad (14.12)$$

where $\delta\mathbf{p}_b$ is defined by (12.50). The INS system model is described in Section 12.2, while dead-reckoning system models are discussed in Section 14.3. The system models for the aiding sensors are the same as for total-state integration.

By analogy with total-state integration, the measurement innovation vector and measurement matrix for each sensor are of the form

$$\delta\mathbf{z}_i^- = \tilde{\mathbf{m}}_i - \delta\hat{\mathbf{m}}_i - \hat{\mathbf{m}}_{\text{Re } f} \quad (14.13)$$

and

$$\mathbf{H}_i^\gamma = \begin{pmatrix} -\frac{\partial \mathbf{m}}{\partial \mathbf{x}_{\text{Re } f}^\gamma} & 0 & \frac{\partial \mathbf{m}}{\partial \mathbf{x}_{\text{Sensor-}i}} & 0 \end{pmatrix}_{\mathbf{x}^\gamma = \mathbf{x}^\gamma} \quad (14.14)$$

where $\hat{\mathbf{m}}_{Ref}$ is the value of \mathbf{m} estimated from the corrected reference navigation solution. Time synchronization may be performed by using a stored reference navigation solution at the measurement time of validity, as described in Section 3.3.4. Measurements with different times of validity may be processed together, providing the appropriate reference solution is used in each case. The measurement models for the aiding sensors are the same as for total-state integration. For position and velocity errors, there is a sign change. Thus,

$$\frac{\partial \mathbf{m}}{\partial \delta \mathbf{v}_{\beta\alpha}^\gamma} = -\frac{\partial \mathbf{m}}{\partial \mathbf{v}_{\beta\alpha}^\gamma}, \quad \frac{\partial \mathbf{m}}{\partial \delta \mathbf{r}_{\beta\alpha}^\gamma} = -\frac{\partial \mathbf{m}}{\partial \mathbf{r}_{\beta\alpha}^\gamma}, \quad \frac{\partial \mathbf{m}}{\partial \delta \mathbf{p}_b} = -\frac{\partial \mathbf{m}}{\partial \mathbf{p}_b} \quad (14.15)$$

Following a measurement update, a dead-reckoning reference navigation solution is corrected in the same way as an inertial navigation solution, as described in Section 12.1.1, with both open- and closed-loop correction options available.

Processing measurement data out of time-of-validity order is not a problem in an error-state Kalman filter, provided the time of validity of the reference navigation solution does not lag behind that of any aiding sensor. However, the problem, described in Section 3.3.4, of repeated application of the same closed-loop corrections to the reference navigation solution can be magnified in a multisensor system where different sensors exhibit different duty cycles and processing lags. Thus, careful Kalman filter design is needed to maintain stability.

As discussed in Section 12.1.1, total-state Kalman filters that incorporate the navigation equations of a reference navigation system in their system models are mathematically equivalent to the sum of the corresponding error-state Kalman filter and reference system navigation equations. Therefore, they are not discussed separately here.

14.2 Terrestrial Radio Navigation

Terrestrial radio navigation systems (Chapter 9) may be incorporated as constituent navigation systems in total-state integration or aiding systems in error-state integration, but not as the reference in error-state integration.

Integration architectures for ranging-based terrestrial radio navigation are essentially the same as for GNSS (see Section 12.1). In loosely coupled integration, position measurements are used, while in tightly coupled integration, pseudo-range, two-way-range, or time-difference measurements are used. Both are described here. Loosely coupled integration is a cascaded architecture where the radio navigation system computes a filtered navigation and a centralized architecture where a single-point position solution is used. Tightly coupled integration is always centralized. The integrated navigation solution may be used to aid radio navigation acquisition and tracking, as with GNSS. Deep integration is also possible, though has yet to be implemented in practice.

Integration of WLAN feature-matching measurements is covered in Section 14.4.

14.2.1 Loosely Coupled Integration

Loosely coupled integration has the advantage of being able to operate with any radio-navigation user equipment. However, measurements are only available where there are sufficient signals to generate a position solution.

Most radio-navigation position solutions exhibit a slowly varying bias, \mathbf{b}_R . Where the integrated navigation system includes a more accurate positioning system, such as GNSS, this bias may be calibrated by estimating it as Kalman filter states. For example, this can significantly improve the accuracy of Loran [7]. Otherwise, it should be modeled as time-correlated measurement noise, as discussed in Section 3.4.2. The bias will exhibit step changes each time the signals used to calculate the navigation solution change. The Kalman filter should respond by increasing the uncertainty of the bias or position states.

A curvilinear position measurement innovation is

$$\delta \mathbf{z}_{R,k}^- = (\tilde{\mathbf{p}}_{aR} - \hat{\mathbf{p}}_b - \hat{\mathbf{T}}_r^p \hat{\mathbf{C}}_b^n \mathbf{l}_{ba}^b - \hat{\mathbf{b}}_R^n)_k \quad (14.16)$$

where k denotes the iteration, $\tilde{\mathbf{p}}_{aR} = (\tilde{L}_{aR}, \tilde{\lambda}_{aR}, \tilde{b}_{aR})$ is the terrestrial-radio-navigation position solution, $\hat{\mathbf{p}}_b = (\hat{L}_b, \hat{\lambda}_b, \hat{b}_b)$ is the integrated position solution at the same time of validity, \mathbf{l}_{ba}^b is the body-to-antenna lever arm, and $\hat{\mathbf{T}}_r^p$ is given by (12.81). Many terrestrial radio navigation systems omit the height component, while others may resolve position in a local grid.

Defining the state vector in terms of curvilinear position as

$$\mathbf{x}^n = \begin{pmatrix} \mathbf{p}_b \\ \vdots \\ \mathbf{b}_R^n \end{pmatrix} \quad (14.17)$$

for total-state integration or

$$\mathbf{x}^n = \begin{pmatrix} \delta \mathbf{p}_b \\ \vdots \\ \mathbf{b}_R^n \end{pmatrix} \quad (14.18)$$

for error-state integration, the measurement matrix is

$$\mathbf{H}_{R,k} = (k_R \mathbf{I} \quad \mathbf{0} \quad \mathbf{I}) \quad (14.19)$$

where k_R is 1 for total-state integration and -1 for error-state integration. The same measurement matrix is obtained where the measurement innovation, position/position error, and bias states are all expressed in terms of ECI or ECEF-frame Cartesian position.

14.2.2 Tightly Coupled Integration

Tightly coupled integration has the advantage of being able to incorporate terrestrial radio navigation measurements when there are insufficient signals to calculate a

single-system position solution. This is particularly useful where a position solution can be obtained by combining range measurements from different radio navigation systems. A further benefit is that the biases are easier to model in the range domain than the position domain [7, 8].

For pseudo-range measurements from passive ranging or range measurements from two-way ranging, the measurement innovation is

$$\delta \mathbf{z}_{R,k}^- = \begin{pmatrix} \tilde{\rho}_{C1} - \hat{\rho}_{C1}^- \\ \tilde{\rho}_{C2} - \hat{\rho}_{C2}^- \\ \vdots \\ \tilde{\rho}_{Cn} - \hat{\rho}_{Cn}^- \end{pmatrix}_k \quad (14.20)$$

where n is the number of transmitters used, $\tilde{\rho}_{Cj}$ is the corrected measured pseudo-range/range for transmitter or tracking channel j , and $\hat{\rho}_{Cj}^-$ is the estimate thereof, calculated from the integrated position solution, transmitter position, and the range bias and clock offset estimates where appropriate. In tightly coupled systems, the receiver clock can be shared between different radio navigation systems.

For delta ranges, $\Delta \tilde{\rho}_{Cij}$, obtained from TDs (see Section 9.2.4), the measurement innovation is

$$\delta \mathbf{z}_{\Delta R,k}^- = \begin{pmatrix} \Delta \tilde{\rho}_{Cm1} - \hat{\rho}_{C1}^- + \hat{\rho}_{Cm}^- \\ \Delta \tilde{\rho}_{Cm2} - \hat{\rho}_{C2}^- + \hat{\rho}_{Cm}^- \\ \vdots \\ \Delta \tilde{\rho}_{Cmn} - \hat{\rho}_{Cn}^- + \hat{\rho}_{Cm}^- \end{pmatrix}_k \quad (14.21)$$

where m is the reference transmitter, noting that this does not have to be common across all measurements.

Let the state vector be defined as

$$\mathbf{x}^\gamma = \begin{pmatrix} \mathbf{r}_{\gamma b}^\gamma \\ \vdots \\ \mathbf{b}_\rho \\ \delta \rho_{rc} \\ \delta \dot{\rho}_{rc} \end{pmatrix} \quad \gamma \in i, e, \quad \mathbf{x}^n = \begin{pmatrix} \mathbf{p}_b \\ \vdots \\ \mathbf{b}_\rho \\ \delta \rho_{rc} \\ \delta \dot{\rho}_{rc} \end{pmatrix} \quad (14.22)$$

for total-state integration or

$$\mathbf{x}^\gamma = \begin{pmatrix} \delta \mathbf{r}_{\gamma b}^\gamma \\ \vdots \\ \mathbf{b}_\rho \\ \delta \rho_{rc} \\ \delta \dot{\rho}_{rc} \end{pmatrix} \quad \gamma \in i, e, \quad \mathbf{x}^n = \begin{pmatrix} \delta \mathbf{p}_b \\ \vdots \\ \mathbf{b}_\rho \\ \delta \rho_{rc} \\ \delta \dot{\rho}_{rc} \end{pmatrix} \quad (14.23)$$

for error-state integration, where \mathbf{b}_ρ are the range biases, $\delta\rho_{rc}$ the clock offset, and $\delta\dot{\rho}_{rc}$ the clock drift. The measurement matrix for pseudo-range measurements is approximately

$$\mathbf{H}_{R,k}^\gamma \approx \begin{pmatrix} -k_R \mathbf{u}_{at,1}^\gamma & 0 & \mathbf{h}_{b,1}^\top & 1 & 0 \\ -k_R \mathbf{u}_{at,2}^\gamma & 0 & \mathbf{h}_{b,2}^\top & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -k_R \mathbf{u}_{at,n}^\gamma & 0 & \mathbf{h}_{b,n}^\top & 1 & 0 \end{pmatrix}_{\mathbf{x}^\gamma = \hat{\mathbf{x}}_k^{\gamma-}} \quad (14.24)$$

or

$$\mathbf{H}_{R,k}^n \approx \begin{pmatrix} -k_R \mathbf{h}_{\rho p,1}^\top & 0 & \mathbf{h}_{b,1}^\top & 1 & 0 \\ -k_R \mathbf{h}_{\rho p,2}^\top & 0 & \mathbf{h}_{b,2}^\top & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ -k_R \mathbf{h}_{\rho p,n}^\top & 0 & \mathbf{h}_{b,n}^\top & 1 & 0 \end{pmatrix}_{\mathbf{x}^\gamma = \hat{\mathbf{x}}_k^{\gamma-}} \quad (14.25)$$

where $\mathbf{u}_{at,j}^\gamma$ is the line-of-sight vector from the antenna to transmitter j ,

$$\mathbf{h}_{\rho p,j} = \begin{pmatrix} (R_N(\hat{L}_b) + \hat{b}_b) u_{at,j,N}^n \\ (R_E(\hat{L}_b) + \hat{b}_b) \cos \hat{L}_b u_{at,j,E}^n \\ -u_{at,j,D}^n \end{pmatrix} \quad (14.26)$$

and

$$\mathbf{h}_{b,j}^\top = (\delta_{1j} \quad \delta_{2j} \quad \dots \quad \delta_{nj}) \quad (14.27)$$

where δ is the Kronecker delta function. For two-way ranging, the clock states are omitted.

For delta-range measurements with a common reference, m , the state vector becomes

$$\mathbf{x}^\gamma = \begin{pmatrix} \mathbf{r}_{\gamma b}^\gamma \\ \vdots \\ \mathbf{b}_\rho \\ b_m \end{pmatrix} \quad \gamma \in i, e, \quad \mathbf{x}^n = \begin{pmatrix} \mathbf{p}_b \\ \vdots \\ \mathbf{b}_\rho \\ b_m \end{pmatrix} \quad (14.28)$$

for total-state integration or

$$\mathbf{x}^\gamma = \begin{pmatrix} \delta \mathbf{r}_{\gamma b}^\gamma \\ \vdots \\ \mathbf{b}_\rho \\ b_m \end{pmatrix} \quad \gamma \in i, e, \quad \mathbf{x}^n = \begin{pmatrix} \delta \mathbf{p}_b \\ \vdots \\ \mathbf{b}_\rho \\ b_m \end{pmatrix} \quad (14.29)$$

for error-state integration. The measurement matrix is then

$$\mathbf{H}_{\Delta R, k}^{\gamma} \approx \begin{pmatrix} -k_R(\mathbf{u}_{at,1}^{\gamma} - \mathbf{u}_{at,m}^{\gamma})^T & 0 & \mathbf{h}_{b,1}^T & -1 \\ -k_R(\mathbf{u}_{at,2}^{\gamma} - \mathbf{u}_{at,m}^{\gamma})^T & 0 & \mathbf{h}_{b,2}^T & -1 \\ \vdots & \vdots & \vdots & \vdots \\ -k_R(\mathbf{u}_{at,n}^{\gamma} - \mathbf{u}_{at,m}^{\gamma})^T & 0 & \mathbf{h}_{b,n}^T & -1 \end{pmatrix} \quad \gamma \in i, e \quad \mathbf{x}^{\gamma} = \hat{\mathbf{x}}_k^{\gamma-} \quad (14.30)$$

or

$$\mathbf{H}_{\Delta R, k}^n \approx \begin{pmatrix} -k_R(\mathbf{h}_{pp,1} - \mathbf{h}_{pp,m})^T & 0 & \mathbf{h}_{b,1}^T & -1 \\ -k_R(\mathbf{h}_{pp,2} - \mathbf{h}_{pp,m})^T & 0 & \mathbf{h}_{b,2}^T & -1 \\ \vdots & \vdots & \vdots & \vdots \\ -k_R(\mathbf{h}_{pp,n} - \mathbf{h}_{pp,m})^T & 0 & \mathbf{h}_{b,n}^T & -1 \end{pmatrix} \quad \mathbf{x}^{\gamma} = \hat{\mathbf{x}}_k^{\gamma-} \quad (14.31)$$

14.3 Dead Reckoning, Attitude, and Height Measurement

Dead-reckoning systems (Chapter 10) may be integrated as either reference or aiding navigation systems (see Section 14.1). The system models for the sensor errors are largely common to the two approaches. Where an INS is used, that should always be the reference, with DR integrated as aiding systems. Otherwise, dead reckoning can provide a reference solution in error-state integration or be a constituent navigation system in total-state integration, integrated in the same way as an aiding system in error-state integration.

A dead-reckoning reference must incorporate both an attitude or heading measurement system and a distance traveled or velocity measurement system. Where the velocity/distance measurement is horizontal only, the reference may also include a height measurement. However, where multiple DR systems provide measurements of the same type (i.e., angular or linear), only one can be the reference.

Attitude and velocity/distance measurements from aiding sensors may be combined to provide a velocity/distance measurement resolved about the ECI, ECEF, or local navigation frame, as described in Sections 10.3 to 10.5. However, treating these as separate measurements allows integrity monitoring, such as measurement innovation filtering (Section 15.3.1), to be performed separately so that measurements from one sensor may be accepted when those from the other are rejected.

Dead reckoning may be used to aid GNSS acquisition and tracking in the same manner as inertial navigation, as described in Section 12.1.4.

Integration of attitude measurements is described first, followed by height measurements, odometers, PDR, and Doppler radar/sonar. Integration of other DR sensors follows similar principles.

14.3.1 Attitude

Where the sensors in an AHRS (Section 10.1.4) or integrated heading measurement system (Section 10.1.3) incorporating a gyro are integrated by Kalman filter, the gyro(s) should be treated as the reference system, with the other sensors integrated as aiding measurements. The attitude error, $\delta\psi_{yb}^\gamma$, and gyro biases, \mathbf{b}_g , should be modeled in the same way as for an INS (see Section 12.2). Where a magnetic compass and a differential odometer, without gyros, are integrated using a Kalman filter, the odometer should be the reference.

An AHRS attitude solution or integrated heading measurement may be integrated with other navigation sensors either as a single unit, forming a cascaded architecture (Section 14.1.2), or as its constituent sensors within a centralized architecture (Section 14.1.3). The centralized architecture enables more representative attitude error modeling but can require more processing power.

14.3.1.1 Leveling

The AHRS leveling measurement innovations (see Section 10.1.1) comprise the difference between the corrected accelerometer-indicated (subscript *A*) and gyro-indicated (no subscript) roll and pitch:

$$\delta\mathbf{z}_{L,k}^- = \begin{pmatrix} \hat{\phi}_{nbA} - \hat{\phi}_{nb} \\ \hat{\theta}_{nbA} - \hat{\theta}_{nb} \end{pmatrix}_k \quad (14.32)$$

where, from (5.89), the leveling measurements may be corrected with estimates of the accelerometer bias, \mathbf{b}_a , using

$$\begin{aligned} \hat{\theta}_{nbA} &= \arctan \left[\frac{-(\tilde{f}_{ib,x}^b - \hat{b}_{a,x})}{\sqrt{(\tilde{f}_{ib,y}^b - \hat{b}_{a,y})^2 + (\tilde{f}_{ib,z}^b - \hat{b}_{a,z})^2}} \right] \\ \hat{\phi}_{nbA} &= \arctan2[-(\tilde{f}_{ib,y}^b - \hat{b}_{a,y}), -(\tilde{f}_{ib,z}^b - \hat{b}_{a,z})] \end{aligned} \quad (14.33)$$

The system model for the accelerometer biases is the same as for an INS (see Section 12.2). To observe them, the AHRS must be swung about the roll and pitch axes and the gyro biases must not be too large.

Defining the AHRS state vector as

$$\mathbf{x}^\gamma = \begin{pmatrix} \delta\psi_{yb}^\gamma \\ \mathbf{b}_g \\ \mathbf{b}_a \\ \mathbf{x}_M \end{pmatrix} \quad \gamma \in i, e, n \quad (14.34)$$

where \mathbf{x}_M comprises the magnetic compass errors, the measurement matrix is

$$\mathbf{H}_{L,k}^{\gamma} = \begin{pmatrix} -\frac{\partial \tilde{\phi}_{nb}}{\partial \delta \psi_{\gamma b}^{\gamma}} & 0 & \frac{\partial \tilde{\phi}_{nbA}}{\partial \mathbf{b}_a} & 0 \\ -\frac{\partial \tilde{\theta}_{nb}}{\partial \delta \psi_{\gamma b}^{\gamma}} & 0 & \frac{\partial \tilde{\theta}_{nbA}}{\partial \mathbf{b}_a} & 0 \end{pmatrix}_{\mathbf{x}^{\gamma} = \hat{\mathbf{x}}_k^{\gamma^-}} \quad (14.35)$$

Where the small angle approximation applies,

$$\begin{pmatrix} \frac{\partial \tilde{\phi}_{nb}}{\partial \delta \psi_{\gamma b}^{\gamma}} \\ \frac{\partial \tilde{\theta}_{nb}}{\partial \delta \psi_{\gamma b}^{\gamma}} \end{pmatrix}_{\mathbf{x}^{\gamma} = \hat{\mathbf{x}}_k^{\gamma^-}} \approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \hat{\mathbf{C}}_{\gamma}^b \quad (14.36)$$

Where an AHRS is integrated with a positioning system, such as GNSS, the position measurements may be used to estimate an acceleration correction to the leveling measurement [9]. However, as pitch and roll correction is an inherent part of the integration of inertial navigation with positioning systems (see Sections 12.2.1 and 12.3.4), a more efficient solution is simply to process low-grade IMU measurements in INS mode when positioning measurements are available. AHRS leveling measurements are then used only during positioning system outages as a reversionary mode [10].

14.3.1.2 Magnetic Heading

Where a magnetic compass (Section 10.1.2) is integrated as an aiding sensor using a Kalman filter, the measurement innovation is

$$\delta z_M^{\gamma} = (\tilde{\psi}_{mb} - \hat{\delta \psi}_{mb} + \hat{\alpha}_{nE} - \hat{\psi}_{nb})_k \quad (14.37)$$

where $\tilde{\psi}_{mb}$ is the magnetic heading; $\hat{\delta \psi}_{mb}$ is the magnetic heading error, estimated from the magnetic-compass Kalman filter states; \mathbf{x}_M , $\hat{\alpha}_{nE}$ is the declination angle of the Earth's magnetic field, obtained from a magnetic model; and $\hat{\psi}_{nb}$ is the integrated heading solution, comprising the corrected INS, AHRS, gyro, or differential-odometer-indicated heading.

If the state vector is defined as

$$\mathbf{x}^{\gamma} = \begin{pmatrix} \delta \psi_{\gamma b}^{\gamma} \\ \vdots \\ \mathbf{x}_M \end{pmatrix} \quad \gamma \in i, e, n \quad (14.38)$$

the measurement matrix is

$$\mathbf{H}_{M,k}^{\gamma} = \left(-\frac{\partial \tilde{\psi}_{nb}}{\partial \delta \psi_{\gamma b}^{\gamma}} \quad 0 \quad \frac{\partial \tilde{\psi}_{mb}}{\partial \mathbf{x}_M} \right)_{\mathbf{x}^{\gamma} = \hat{\mathbf{x}}_k^{\gamma^-}} \quad (14.39)$$

Where the small angle approximation applies,

$$\frac{\partial \tilde{\psi}_{nb}}{\partial \delta \boldsymbol{\psi}_{\gamma b}^{\gamma}} \Big|_{\mathbf{x}^{\gamma} = \hat{\mathbf{x}}_k^{\gamma}} \approx (0 \quad 0 \quad 1) \hat{\mathbf{C}}_{\gamma}^b \quad (14.40)$$

The choice of magnetometer errors to estimate as Kalman filter states varies with the application. If the hard- and soft-iron equipment magnetism is adequately calibrated by the magnetic compass software and does not vary after that calibration, no further calibration is needed in the integration algorithm. Otherwise, the Kalman filter may estimate body-frame hard- and soft-iron magnetism, \mathbf{b}_m and \mathbf{M}_m , as defined by (10.10), or heading-domain hard- and soft-iron calibration coefficients, c_{h1} , c_{h2} , c_{s1} , and c_{s2} , as defined by (10.11). Often, only the hard-iron magnetism need be estimated. To observe the heading-domain equipment-magnetism coefficients, the host vehicle must perform turns, while to observe the full body-frame magnetism, maneuvers about the roll and pitch axes are also required. Thus, for marine and land vehicle applications, the heading-domain model is better.

The Kalman filter may also estimate regional and temporal deviations in the declination angle from that given by the magnetic model. To observe this, the observability of the heading error from positioning measurements through the growth in the velocity error must be strong (see Sections 12.2.1 and 12.3.4).

14.3.2 Height and Depth

Barometric, radar, or sonar height and depth measurements (Section 10.2) may be integrated as reference or aiding sensors. Where they form the reference, their systematic errors are accounted for by the navigation solution height and vertical velocity error states of the Kalman filter. Where they are integrated as aiding sensors, dedicated error states may be required. For a baro or depth pressure sensor operating near the surface, only the bias need be estimated. Otherwise, a scale factor error should also be estimated.

For a radar altimeter, error states are not needed unless the terrain height database is biased. However, the assumed measurement noise covariance, R_B , should be varied with the radar footprint size and terrain roughness. For sonar altimeter measurements, estimation of a scale factor error can account for errors in the assumed speed of sound.

For a baro integrated as an aiding sensor with a bias, b_b , and scale factor error, s_b , estimated, the measurement innovation is

$$\delta z_{B,k}^- = [\tilde{h}_{B,B} - \hat{h}_b - \hat{h}_b(1 + \hat{s}_b)]_k \quad (14.41)$$

where $\tilde{h}_{B,B}$ is the barometric height measurement and \hat{h}_b is the integrated geodetic height solution. If the state vector is $\mathbf{x} = (\delta h_b, \dots, b_b, s_b)$, the measurement matrix is then

$$\mathbf{H}_{B,k} \approx (-1 \quad 0 \quad 1 \quad \hat{h}_{b,k}^-) \quad (14.42)$$

neglecting the products of the error states.

14.3.3 Odometers

Integration of odometer measurements (Section 10.3) requires the estimation of the scale factor errors as Kalman filter states. Where the speed of individual wheels is measured, a scale factor error state is needed for each wheel used. These may be attributed to each wheel as s_{orL} , s_{orR} , s_{ofL} , and s_{ofR} , where r denotes rear, f forward, L left, and R right. Alternatively, an average and difference scale factor error may be estimated for each pair:

$$s_{or} = \frac{1}{2}(s_{orL} + s_{orR}), \quad s_{\Delta or} = s_{orL} - s_{orR} \quad (14.43)$$

$$s_{of} = \frac{1}{2}(s_{ofL} + s_{ofR}), \quad s_{\Delta of} = s_{ofL} - s_{ofR}$$

As tire wear is the main long-term cause of scale factor variation and heating is the main short-term cause, the scale factor errors of each pair of wheels will be highly correlated. For single-wheel scale factor states, this must be modeled with nonzero off-diagonal elements of the initial error covariance and system noise covariance matrices, P_0^+ and Q . The average and difference scale factor error states are uncorrelated. The initial uncertainties and system noise should be larger for the average states. Where transmission-shaft measurements are used, only s_{or} or s_{of} is estimated.

Where odometers are integrated as the reference navigation system, or part thereof, the effect of the scale factor errors on the navigation solution is represented within the system model [11]. Considering rear wheel sensors only with s_{or} and $s_{\Delta or}$ estimated, the state dynamics are

$$\delta \dot{\mathbf{r}}_{\gamma b}^\gamma = s_{or} \mathbf{C}_n^\gamma \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{C}_\gamma^n \mathbf{v}_{\gamma b}^\gamma \quad \gamma \in i, e \quad (14.44)$$

or

$$\delta \dot{L}_b = \frac{s_{or} v_{eb,N}^n}{R_N(L_b) + h_b} \quad (14.45)$$

$$\delta \dot{\lambda}_b = \frac{s_{or} v_{eb,E}^n}{[R_E(L_b) + h_b] \cos L_b}$$

and, from (10.26),

$$\delta \dot{\psi}_{\gamma b}^\gamma = \left(s_{or} \dot{\psi}_{nb} + s_{\Delta or} \frac{v_{er}}{T_r} \right) \mathbf{C}_n^\gamma \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad (14.46)$$

where T_r is the rear track width, R_N and R_E are given by (2.65) and (2.66), and it is assumed that odometer measurements are only used in the horizontal plane.

Where odometer measurements are incorporated as aiding sensors or total-state integration is used, there are a number of different ways of processing individual wheel-speed measurement. They may be combined into a velocity or distance traveled in the local navigation, ECEF, or ECI frame. They may be expressed as a speed or distance traveled in the body-frame forward direction, together with a heading rate or change. Alternatively, separate measurements for each wheel may be processed. Each provides the same information. Where a transmission-shaft sensor is used, only the forward speed or distance traveled is available [12].

In all cases, the assumption of zero velocity along each wheel axle is commonly made. This is an example of a nonholonomic constraint. Instantaneous odometer velocity measurements are noisy, so if velocity, rather than distance, is used, it should be averaged over the time since the last odometer measurement.

Where the odometer measurements comprise the rear-wheel forward speed, \tilde{v}_{erO} , and yaw rate, $\tilde{\psi}_{nbO}$, averaged over the interval $t - \tau_o$ to t , together with the assumption of a zero cross-track speed, the measurement innovation is

$$\delta \bar{\mathbf{z}}_{O,k} = \begin{pmatrix} \tilde{v}_{erO}(1 - \hat{s}_{or}) - \hat{v}_{er} \\ \tilde{\psi}_{nbO}(1 - \hat{s}_{or}) - \frac{\hat{v}_{er}}{T_r} \hat{s}_{\Delta or} - \hat{\psi}_{nb} \\ \hat{v}_x \end{pmatrix}_k \quad (14.47)$$

where \hat{v}_{er} , $\hat{\psi}_{nb}$, and \hat{v}_x are, respectively, the predicted rear-wheel forward speed, yaw rate, and cross-track speed from the corrected navigation solution. They are obtained from a store of the reference velocity and attitude using

$$\hat{v}_{er} = \frac{1}{\tau_o} \int_{t-\tau_o}^t \cos \hat{\theta}_{nb} (1 \ 0 \ 0) [\hat{\mathbf{C}}_\gamma^b(t') \hat{\mathbf{v}}_{\beta b}^\gamma(t') + (\boldsymbol{\omega}_{\beta b}^b(t') \wedge \mathbf{l}_{br}^b)] dt' \quad \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (14.48)$$

$$\hat{\psi}_{nb} = \frac{1}{\tau_o} [\hat{\psi}_{nb}(t) - \hat{\psi}_{nb}(t - \tau_o)] \quad (14.49)$$

and

$$\hat{v}_x = \frac{1}{\tau_o} \int_{t-\tau_o}^t (0 \ 1 \ 0) [\hat{\mathbf{C}}_\gamma^b(t') \hat{\mathbf{v}}_{\beta b}^\gamma(t') + (\boldsymbol{\omega}_{\beta b}^b(t') \wedge \mathbf{l}_{br}^b)] dt' \quad \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (14.50)$$

where it is assumed that a pitch correction, $\cos \theta_{nb}$, has been applied to the odometer measurements, as described in Section 10.3.

Defining the state vector as

$$\mathbf{x}^\gamma = \begin{bmatrix} \delta\psi_{\gamma b}^\gamma \\ \delta v_{\beta b}^\gamma \\ \vdots \\ \mathbf{b}_g \\ \vdots \\ s_{or} \\ \delta\Delta_{or} \end{bmatrix} \quad \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (14.51)$$

noting that the gyro bias is only estimated where the reference navigation system includes gyros, the measurement matrix is

$$\mathbf{H}_{O,k}^\gamma = \begin{bmatrix} -\frac{\partial \hat{v}_{er}}{\partial \delta\psi_{\gamma b}^\gamma} & -\frac{\partial \hat{v}_{er}}{\partial \delta v_{\beta b}^\gamma} & 0 & 0 & 0 & \hat{v}_{er} & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau_o} (0 \ 0 \ 1) \hat{\mathbf{C}}_b^n & 0 & \hat{\psi}_{nb} & \frac{\hat{v}_{er}}{T_r} \\ -\frac{\partial \hat{v}_x}{\partial \delta\psi_{\gamma b}^\gamma} & -\frac{\partial \hat{v}_x}{\partial \delta v_{\beta b}^\gamma} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{x}^\gamma = \hat{\mathbf{x}}_k^{\gamma-} \quad (14.52)$$

Integration of front-wheel odometer measurements is more complex, as the front-wheel frame is displaced from the body frame by the steering angle, ψ_{bf} , as shown in Section 10.3. The integration algorithm must thus input the steering angle and rate thereof as well as the odometer measurements. The steering-angle bias and scale factor may be estimated as Kalman filter states where necessary.

14.3.4 Pedestrian Dead Reckoning

Integration of PDR measurements (see Section 10.4) requires the estimation of the coefficients of the step length estimation model, \mathbf{c}_P , or corrections to their default values, together with the boresight angle, ψ_{bb} , where this is unknown or variable. Where PDR is integrated as part of the reference navigation system, the effect of these PDR states on the navigation solution is represented within the system model.

Where PDR is integrated as an aiding measurement or in a total-state architecture, the measurement may be resolved in either the body frame or the frame used for resolving the position solution. A nonholonomic constraint of zero sideways motion may also be applied. The measurement innovation in the projection of the body frame into the horizontal plane, applying the sideways motion constraint, is

$$\delta \mathbf{z}_{P,k}^- = \begin{pmatrix} \Delta r_{PDR}(\hat{\mathbf{c}}_P) \cos \hat{\psi}_{bb} - \Delta r'_x \\ \Delta r_{PDR}(\hat{\mathbf{c}}_P) \sin \hat{\psi}_{bb} - \Delta r'_y \end{pmatrix}_k \quad (14.53)$$

where

$$\begin{pmatrix} \Delta r'_x \\ \Delta r'_y \end{pmatrix} = \int_{t-\tau_p}^t \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \hat{\mathbf{C}}_n^b(t') \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \hat{\mathbf{C}}_\gamma^n(t') \hat{\mathbf{v}}_{\beta b}^\gamma(t') dt' \\ \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (14.54)$$

and τ_p is the duration of the step.

Defining the state vector as

$$\mathbf{x}^\gamma = \begin{pmatrix} \delta\psi_{\gamma b}^\gamma \\ \delta\mathbf{v}_{\beta b}^\gamma \\ \vdots \\ \mathbf{c}_P \\ \psi_{bb} \end{pmatrix} \quad \{\beta, \gamma\} \in \{i, i\}, \{e, e\}, \{e, n\} \quad (14.55)$$

the measurement matrix is

$$\mathbf{H}_{P,k}^\gamma = \begin{pmatrix} -\frac{\partial \Delta r'_x}{\partial \delta\psi_{\gamma b}^\gamma} & -\frac{\partial \Delta r'_x}{\partial \delta\mathbf{v}_{\beta b}^\gamma} & 0 & \cos \hat{\psi}_{bb} \frac{\partial \Delta r_{PDR}}{\partial \mathbf{c}_P} & -\sin \hat{\psi}_{bb} \Delta r_{PDR}(\hat{\mathbf{c}}_P) \\ -\frac{\partial \Delta r'_y}{\partial \delta\psi_{\gamma b}^\gamma} & -\frac{\partial \Delta r'_y}{\partial \delta\mathbf{v}_{\beta b}^\gamma} & 0 & \sin \hat{\psi}_{bb} \frac{\partial \Delta r_{PDR}}{\partial \mathbf{c}_P} & \cos \hat{\psi}_{bb} \Delta r_{PDR}(\hat{\mathbf{c}}_P) \end{pmatrix} \quad \mathbf{x}^\gamma = \hat{\mathbf{x}}_k^{\gamma-} \quad (14.56)$$

14.3.5 Doppler Radar and Sonar

Integration of Doppler radar and sonar measurements (see Section 10.5) requires estimation of a single scale factor error state, s_D , by the Kalman filter. For radar, this accounts for the effect of scattering over water, while for sonar, it accounts for errors in the assumed speed of sound. For radar over water, the surface velocity is also estimated where measurements from a positioning system, such as GNSS, are available. Further states may be estimated if the beam alignment is unknown.

Where Doppler is integrated as part of the reference navigation system, the effect of the scale factor error on the navigation solution is accounted for in the system model. The state dynamics are

$$\delta\dot{\mathbf{r}}_{\gamma b}^\gamma = s_D \mathbf{v}_{\gamma b}^\gamma \quad \gamma \in i, e \quad (14.57)$$

or

$$\begin{aligned} \partial\dot{L}_b &= \frac{s_D v_{eb,N}^n}{R_N(L_b) + h_b} \\ \partial\dot{\lambda}_b &= \frac{s_D v_{eb,E}^n}{[R_E(L_b) + h_b] \cos L_b} \\ \partial\dot{h}_b &= -s_D v_{eb,D}^n \end{aligned} \quad (14.58)$$

Where Doppler is incorporated as an aiding sensor or total-state integration is used, the Doppler velocity may be resolved about the ECI, ECEF, local navigation, or body frame. Alternatively, each beam may be processed as a separate measurement. For a body-frame Doppler velocity measurement, $\tilde{\mathbf{v}}_{ebD}^e$, where the beam alignment is known, the measurement innovation is

$$\delta \mathbf{z}_{D,k}^- = [\tilde{\mathbf{v}}_{ebD}^b (1 - \hat{s}_D) - \hat{\mathbf{C}}_\gamma^b \mathbf{v}_{eb}^\gamma]_k \quad \gamma \in e, n \quad (14.59)$$

Defining the state vector as

$$\mathbf{x}^\gamma = \begin{pmatrix} \delta\psi_{\gamma b}^\gamma \\ \delta\mathbf{v}_{eb}^\gamma \\ \vdots \\ s_D \end{pmatrix} \quad \gamma \in e, n \quad (14.60)$$

the measurement matrix is

$$\mathbf{H}_{D,k}^\gamma = (-\hat{\mathbf{C}}_\gamma^b [\hat{\mathbf{v}}_{eb}^\gamma \wedge] \quad -\hat{\mathbf{C}}_\gamma^b \quad 0 \quad -\hat{\mathbf{C}}_\gamma^b \hat{\mathbf{v}}_{eb}^\gamma)_{\mathbf{x}^\gamma = \hat{\mathbf{x}}_k^\gamma} \quad (14.61)$$

14.4 Feature Matching

Feature-matching techniques (Chapter 11), such as terrain-referenced navigation, image matching, map matching, and WLAN positioning, may be incorporated as constituent navigation systems in total-state integration or aiding measurements in error-state integration, but not as the reference in error-state integration. This section describes the integration of position fixes and line fixes, followed by the handling of ambiguous measurement.

14.4.1 Position Fixes

Most feature-matching systems produce two- or three-dimensional curvilinear position fixes, $\tilde{\mathbf{p}}_{ff}$, where f denotes the body frame of the sensor. These position fixes may be subject to a slowly varying database bias, \mathbf{b}_F , and/or sensor alignment errors, $\delta\psi_{bf}^f$, which may be calibrated by estimating them as Kalman filter states in the integration algorithm.

The measurement innovation with a downward-looking (along the z direction of the f frame) sensor on an air vehicle is

$$\delta \mathbf{z}_{F,k}^n = \left\{ \tilde{\mathbf{p}}_{ff} - \hat{\mathbf{p}}_b - \hat{\mathbf{b}}_F^n - \hat{\mathbf{T}}_r^p \left[\hat{\mathbf{C}}_b^n \mathbf{l}_{bf}^b + \Delta h_{tf} \begin{pmatrix} \delta\hat{\psi}_{bf,x}^f \sin \hat{\psi}_{nb} + \delta\hat{\psi}_{bf,y}^f \cos \hat{\psi}_{nb} \\ -\delta\hat{\psi}_{bf,x}^f \cos \hat{\psi}_{nb} + \delta\hat{\psi}_{bf,y}^f \sin \hat{\psi}_{nb} \\ 0 \end{pmatrix} \right] \right\}_k \quad (14.62)$$

where \mathbf{l}_{bf}^b is the body-to-sensor lever arm, $\hat{\mathbf{T}}_r^p$ is given by (12.81), and Δh_{tf} is the height of the sensor above the terrain containing the matched features.

Defining the state vector in terms of curvilinear position as

$$\mathbf{x}^n = \begin{pmatrix} \mathbf{p}_b \\ \vdots \\ \mathbf{b}_F^n \\ \delta\psi_{bf,x}^f \\ \delta\psi_{bf,y}^f \end{pmatrix} \quad (14.63)$$

for total-state integration or

$$\mathbf{x}^n = \begin{pmatrix} \delta\mathbf{p}_b \\ \vdots \\ \mathbf{b}_F^n \\ \delta\psi_{bf,x}^f \\ \delta\psi_{bf,y}^f \end{pmatrix} \quad (14.64)$$

for error-state integration, the measurement matrix is

$$\mathbf{H}_{F,k}^n = \left[k_F \mathbf{I} \quad \mathbf{0} \quad \mathbf{I} \quad \hat{\mathbf{T}}_r^p \Delta h_{tf} \begin{pmatrix} \sin \hat{\psi}_{nb} \\ -\cos \hat{\psi}_{nb} \\ 0 \end{pmatrix} \quad \hat{\mathbf{T}}_r^p \Delta h_{tf} \begin{pmatrix} \cos \hat{\psi}_{nb} \\ \sin \hat{\psi}_{nb} \\ 0 \end{pmatrix} \right]_k \quad (14.65)$$

where k_F is 1 for total-state integration and -1 for error-state integration.

The integration of position fixes is centralized where successive fixes use independent measurements from the feature-matching sensor and cascaded where past information is used in the feature-matching system (e.g., to resolve ambiguous measurements). For centralized integration of a sequential TRN system (Section 11.1.1), the measurement innovations, δz_T , should be input directly to the integration Kalman filter.

Feature-matching systems require the integrated navigation solution to be fed back to them to indicate which region of the database should be searched and, in some cases, to provide velocity or indicate the sensor alignment. Therefore, some systems output position corrections, $\tilde{\mathbf{p}}_F - \hat{\mathbf{p}}_b - \hat{\mathbf{T}}_r^p \hat{\mathbf{C}}_b^n \mathbf{l}_{bf}^b$, instead of fixes.

14.4.2 Line Fixes

Some feature matching systems, such as continuous visual navigation and map matching, can sometimes determine that a user is somewhere along a line, but not where along the line they are, providing a one-dimensional measurement in the

horizontal plane. This may be specified in terms of an arbitrary point on that line, $\tilde{\mathbf{p}}_{ff}$, and the bearing of the line, $\tilde{\psi}_{nF}$. The perpendicular distance from the estimated position, $\hat{\mathbf{p}}_b$, to the line forms the measurement innovation, δz_F , while the bearing of the line is used in the measurement matrix. Figure 14.12 illustrates this. The lever arm, database bias, and sensor alignment errors are neglected in this discussion.

The measurement innovation may be calculated using

$$\delta z_{F,k} = [(-\sin \tilde{\psi}_{nF} \quad \cos \tilde{\psi}_{nF} \quad 0) \hat{\mathbf{T}}_p^r (\tilde{\mathbf{p}}_{ff} - \hat{\mathbf{p}}_b)]_k \quad (14.66)$$

where

$$\hat{\mathbf{T}}_p^r = (\hat{\mathbf{T}}_r^p)^{-1} = \begin{pmatrix} R_N(\hat{L}_b) + \hat{h}_b & 0 & 0 \\ 0 & (R_E(\hat{L}_b) + \hat{h}_b) \cos \hat{L}_b & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad (14.67)$$

Where the state vector is simply \mathbf{p}_b or $\delta \mathbf{p}_b$, the measurement matrix is then

$$\mathbf{H}_{F,k}^n = k_F [(\sin \tilde{\psi}_{nF} \quad -\cos \tilde{\psi}_{nF} \quad 0) \hat{\mathbf{T}}_p^r]_k \quad (14.68)$$

As feature-matching systems require the integrated position solution, the line fix may instead be output as a position correction, $\delta \tilde{z}_F$, and the bearing of that correction, $\tilde{\psi}'_{nF}$, also shown in Figure 14.12. In this case, the measurement innovation is simply the correction and the measurement matrix is

$$\mathbf{H}_{F,k}^n = -k_F [(\cos \tilde{\psi}'_{nF} \quad \sin \tilde{\psi}'_{nF} \quad 0) \hat{\mathbf{T}}_p^r]_k \quad (14.69)$$

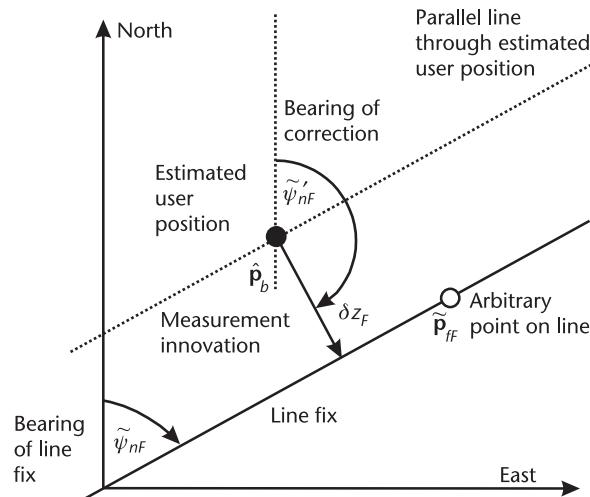


Figure 14.12 Geometry of a line-fix measurement.

14.4.3 Ambiguous Measurements

In feature-matching systems, there is not always a clear match between the measurements and the database. Measurement and database resolution limitations can make it difficult to distinguish nearby features, while environments often contain repeating features. For example, an image-matching system may see multiple rail tracks but only have a single line on its database. Due to these ambiguities, feature-matching systems all produce occasional wrong measurements. To prevent these from corrupting the integrated navigation solution, measurement innovation-based fault detection and integrity-monitoring techniques, described in Section 15.3, should always be used.

A more robust approach is for the feature-matching system to output a multiple-hypothesis measurement, comprising a position, covariance, and probability for each of the most likely matches between the measurement and database. There should also be a null hypothesis that none of the matches is correct. The integration algorithm can use innovation filtering (Section 15.3.1) to reject any hypotheses totally inconsistent with the prior navigation solution and then incorporate the remaining hypotheses using the best-fix, weighted-fix, or multiple-hypothesis filtering method, as described in Section 3.4.4.

With TRN operating over low-roughness terrain with a conventional radalt sensor, using a weighted-fix integration filter makes the navigation solution more robust against false TRN fixes than a simple best-fix approach, but additional improvements have not been obtained with multiple-hypothesis filtering [13]. However, for CVN, where false matches can be correlated over successive images, multiple-hypothesis filtering is a key part of the system [14].

References

- [1] Carlson, N. A., and M. P. Berarducci, "Federated Kalman Filter Simulation Results," *Navigation: JION*, Vol. 41, No. 3, 1994, pp. 297–321.
- [2] Carlson, N. A., "Federated Filter for Distributed Navigation and Tracking Applications," *Proc. ION 58th AM*, Albuquerque, NM, June 2002, pp. 340–353.
- [3] Carlson, N. A., "Federated Square Root Filter for Decentralized Parallel Processing," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 26, No. 3, 1990, pp. 517–525.
- [4] Hamilton, A. S., and B. W. Chilton, "A Flexible Federated Navigation System for Next Generation Military Aircraft," *Proc. ION 52nd AM*, Cambridge, MA, June 1996, pp. 409–415.
- [5] Levy, L. J., "Suboptimality of Cascaded and Federated Kalman Filters," *Proc. ION 52nd AM*, Cambridge, MA, June 1996, pp. 399–407.
- [6] Yunck, T. P., "Orbit Determination," in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 559–592.
- [7] Enge, P. K., and J. R. McCullough, "Aiding GPS with Calibrated Loran-C," *Navigation: JION Navigation*, Vol. 35, No. 4, 1988, pp. 469–482.
- [8] Hide, C., et al., "Integrated GPS, LORAN-C and INS for Land Navigation Applications," *Proc. ION GNSS 2006*, Fort Worth, TX, September 2006, pp. 59–67.
- [9] Gebre-Egziabher, D., et al., "A Gyro-Free Quaternion-Based Attitude Determination System Suitable for Implementation Using Low Cost Sensors," *Proc. IEEE PLANS*, San Diego, CA, March 2000, pp. 185–192.

- [10] Wendel, J., et al., “MAV Attitude Estimation Using Low-Cost MEMS Inertial Sensors and GPS,” *Proc. ION 61st AM*, Cambridge, MA, June 2005, pp. 397–403.
- [11] Carlson, C. R., J. C. Gerdes, and J. D. Powell, “Error Sources When Land Vehicle Dead Reckoning with Differential Wheelspeeds,” *Navigation: JION*, Vol. 51, No. 1, 2004, pp. 13–27.
- [12] Zhao, L., et al., “An Extended Kalman Filter Algorithm for Integrating GPS and Low Cost Dead Reckoning System Data for Vehicle Performance and Emissions Monitoring,” *Journal of Navigation*, Vol. 56, No. 2, 2003, pp. 257–275.
- [13] Groves, P. D., R. J. Handley, and A. R. Runnalls, “Optimising the Integration of Terrain Referenced Navigation with INS and GPS,” *Journal of Navigation*, Vol. 59, No. 1, 2006, pp. 71–89.
- [14] Handley, R. J., L. Dack, and P. McNeil, “Flight Trials of the Continuous Visual Navigation System,” *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 185–192.

Fault Detection and Integrity Monitoring

Like any technology, a navigation system can occasionally produce output errors much larger than the uncertainty bounds specified for or indicated by it. This may be due to hardware or software failures or to unusual operating conditions. Integrity monitoring systems detect these faults and protect the overall navigation solution. They may operate at a number of levels. *Fault detection* simply indicates that a fault is present and the user warned. Fault detection and *recovery* (FDR) identifies where the fault lies and attempts to recover from navigation solution contamination occurring prior to the detection of that fault. Fault detection and *isolation* (FDI) provides a navigation solution that is isolated from (i.e., uncontaminated by) the faulty data that has been detected. Fault detection and *exclusion* (FDE) additionally verifies that the recovered navigation solution is free from faults. Note that some authors define these terms differently.

The highest level of integrity monitoring requires certification that the navigation solution meets a number of performance criteria, such as the availability of a fault-free navigation solution and the probability of failing to identify a fault.

Many of the integrity monitoring techniques described here rely on consistency checks between different measurements. These require redundant information (i.e., more data than is required to form a navigation solution). Fault detection requires at least one redundant measurement, while FDE requires at least two.

Section 15.1 discusses the failure modes that can occur in navigation systems. Section 15.2 discusses fault detection through range checks. Section 15.3 describes Kalman filter innovation monitoring, and Section 15.4 describes integrity monitoring through direct consistency checks between quantities calculated from different combinations of measurements. Finally, Section 15.5 discusses certification that an integrity monitoring system fulfils a required navigation performance (RNP). The focus of the chapter is on user-based integrity monitoring, sometimes described as *sensor level*, rather than infrastructure-based monitoring, sometimes known as *system level*. The different integrity monitoring techniques are not mutually exclusive and are often combined.

15.1 Failure Modes

The main navigation system faults that can occur and should be detected by an integrity monitoring system are summarized for each of the main navigation

technologies. The section concludes with a discussion of integration algorithm failure modes.

15.1.1 Inertial Navigation

Individual inertial sensor faults are due to hardware failure and can manifest as no outputs at all, null readings, repeated readings, or simply much larger errors than specified. Once a sensor fault has been detected, no further data from that sensor should be accepted. Unless there are redundant inertial sensors, this means discarding the whole inertial navigation solution. Large errors exhibited by all the inertial sensors can be an indication of a much-higher vibration environment than the system is designed for or of a mounting failure. The whole IMU or INS may also exhibit a power failure, software failure, or communications failure, in which case a reset should be attempted. Achieving FDR, FDI, or FDE with inertial navigation requires redundant hardware.

15.1.2 GNSS

GNSS failure modes may be divided into four categories: satellite faults; unusual atmospheric propagation; local channel failures, which comprise faults affecting individual channels of a single set of user equipment; and general user equipment faults. A summary may be found in [1].

Satellite faults are best detected through ground-based signal monitoring networks, as described in Section 8.5, an example of system level integrity monitoring. These alert users of failures using the GNSS broadcast navigation messages, SBAS, GBAS, or other communication links.

Large ionospheric or tropospheric propagation delays, or ionospheric scintillation, can occur due to solar and meteorological storms. These may or may not be detected by monitoring networks, depending on the density of the network and the locality of the storm. Storms and weather fronts can also result in larger than usual residual errors in differential and relative GNSS. Thus, unusual atmospheric propagation can manifest as local channel failures.

Other causes of local channel failures are multipath, tracking loops in the process of losing lock, and receiver hardware and software faults affecting individual channels. Satellite faults and wide-area atmospheric problems may be detected as local channel failures where no monitoring network is used or to provide a backup to a network. FDI and FDE level protection against single-channel failures can usually be obtained provided sufficient satellites are tracked. The faulty signal is simply discarded and a navigation solution computed using the remaining signals.

General user equipment faults affect all satellite signals. Hardware failure may occur in the antenna, receiver, or reference oscillator, while the processing and communication functions may be subject to hardware or software failure. These faults can sometimes produce erroneous outputs on all channels, as opposed to no output at all, so they require detection. Software faults can be recovered by resetting the user equipment. Recovery from hardware faults requires redundant hardware.

15.1.3 Terrestrial Radio Navigation

Terrestrial radio navigation systems essentially exhibit the same kinds of faults as GNSS. All systems are liable to transmitter faults, whereby incorrect signals are broadcast, while unusual atmospheric conditions and floods can affect the propagation of Loran signals. Signal monitoring networks are implemented for Loran, but they are much less common for other systems.

All radio signals can be subject to multipath, while interference or signal attenuation can affect all or some signals at a given location. These problems can cause many receivers to produce a stream of false measurements prior to detecting signal unavailability. User equipment can also exhibit hardware or software failures.

15.1.4 Dead Reckoning, Attitude, and Height Measurement

Like any other equipment, dead-reckoning systems are subject to hardware or software failure, while the individual technologies have their own short-term and long-term failure modes. Magnetic heading measurements can exhibit errors due to incorrect calibration of equipment magnetism, environmental magnetic anomalies, and problems determining the sensor orientation under acceleration. Barometric height can exhibit discontinuities in the presence of sonic booms and weather fronts. Odometers are subject to wheel slip, while PDR algorithms may be fooled by unexpected movements. Doppler radar and sonar velocity measurements exhibit errors when a moving animal or vehicle interrupts one of the beams. Image-correlation systems are subject to occasional false matches between successive images. Air data and ship's log measurements are vulnerable to turbulence and other changes in wind velocity and water currents.

15.1.5 Feature Matching

Feature-matching systems are inherently unreliable in that there will always be a possibility of false matches between the measurements and the database. The probability of false fixes may be traded off against the availability of fixes, but false fixes cannot be eliminated altogether. Feature-matching fixes should be validated against earlier fixes and other positioning systems where available. Otherwise, they should be treated as provisional until they can be verified by later fixes, noting that inertial navigation or dead reckoning can be used to aid comparison of successive position fixes. Multiple-hypothesis Kalman filtering techniques are discussed in Section 3.4.4. Arguably, feature matching should not be used at all for applications with very high integrity requirements.

15.1.6 Integration Algorithm

In addition to processor hardware and software failure, there are three ways in which faults can manifest in a Kalman filter-based integration algorithm or navigation processor. These are numerical problems, poor tuning, and model failure. Numerical problems are discussed in Section 3.3.3; they should be designed out but may still occur if the Kalman filter is run for a longer period than it is designed for.

As discussed in Section 3.3.1, Kalman filter tuning is a tradeoff between convergence rate and stability. If the values assumed for the initial uncertainty, system noise, and measurement noise are overoptimistic, the errors in the state estimates are more likely to be significantly larger than the state uncertainties.

Model failures occur where the system and measurement models don't properly account for a navigation system's behavior. For example, higher order inertial sensor errors, receiver clock g-dependent errors, or vibration-induced errors may be erroneously neglected. Alternatively, the user dynamics may be underestimated, time-synchronization errors, or time-correlated noise may be unaccounted for, or the variation in GNSS measurement noise with c/n_0 may be neglected. These can all lead to state-estimation errors significantly exceeding the state uncertainties.

Where state-estimation errors due to poor tuning or modeling can be detected, they may be remedied by increasing the state uncertainties (see Section 15.3.3). However, for applications with high integrity requirements, effort must be expended to model all the error sources and correctly tune the Kalman filter. Integrity monitoring should not be used as a substitute for poor design.

The same failure modes apply to non-Kalman filter-based estimation algorithms.

15.2 Range Checks

This section discusses the application of checks to validate that the sensor outputs, navigation solution(s), and Kalman filter estimates lie within reasonable ranges. Parameters deviating outside their normal operational ranges can be indicative of a fault, though the converse does not apply. Range checks will not detect all failure modes, but sensor output tests will respond immediately to gross faults, protecting the navigation solution, while navigation solution checks provide an extra layer of protection, and Kalman filter estimate checks can detect slow-building faults.

15.2.1 Sensor Outputs

A number of checks may be performed on the navigation sensor outputs, such as accelerometer and gyro measurements, GNSS pseudo-ranges and pseudo-range rates, or I_s and Q_s , magnetic field measurements, and radar ranges. Absolute values and step changes in measurements may be compared against the operational ranges of the sensors as specified by the manufacturer and against the expected operating range of the sensor environment. For example, all vehicles have a maximum acceleration and angular rate. However, higher values may be measured due to vibration, so, in many cases, it is more effective to apply range checks to smoothed measurements. The maximum GNSS pseudo-range rate is a function of the satellite velocity and the ranges of the user velocity and receiver clock drift. Magnetic field measurements may be compared with the Earth's magnetic field strength, though it should be noted that discrepancies are more likely to be due to environmental anomalies than sensor failure, meriting a temporary rather than permanent rejection of measurements.

Failure to produce any measurements or a stream of null measurements is an indication of sensor failure. However, faulty sensors can also produce a succession of repeated measurements. A single repeated measurement should not be treated as a fault, as it can occur by chance or due to a communication glitch. The likelihood of chance repeated measurements depends on the ratio of the sensor noise to the quantization level, so the number of repetitions signifying a fault should be set accordingly.

15.2.2 Navigation Solution

Every navigation system user has an operational envelope. A land vehicle or ship should always be close to the Earth's surface, and every aircraft has a maximum altitude above which it cannot fly. Similarly, every vehicle has a maximum speed with respect to the Earth. For example, civil airliners currently in service do not exceed the speed of sound, and road vehicles rarely exceed 50 m s^{-1} . Therefore, if a navigation system is indicating a position or velocity outside the user's operational envelope, there is probably a fault. This also applies to solutions for the GNSS receiver clock drift that significantly exceed the reference oscillator specification.

15.2.3 Kalman Filter Estimates

The Kalman filter is a powerful tool for detecting faults. In most integrated navigation systems including an IMU, the accelerometer and gyro biases are estimated as states. Therefore, if a bias estimate is several times the standard deviation specified by the manufacturer (a threshold of order 5σ is suitable), then there is likely to be a fault with the sensor. Outlying state estimates can also occur when the INS calibration is poor due to a lack of measurements or observability problems, so the current state uncertainties should be accounted for in any fault-detection test.

Range checks may also be applied to GNSS range-bias state estimates. Where these are modeled with short correlation times, large estimates may indicate multipath errors. A similar approach may be adopted for terrestrial radio navigation, while feature-matching errors can sometimes be detected through large estimates of database biases. Range checks on Kalman filter estimates of magnetic heading bias and barometric altimeter and Doppler radar/sonar scale factor errors can also be used to detect faults.

State estimate checks are essentially a form of consistency check, so they rely on redundant measurement data, though not necessarily of the same type. FDR can be achieved by rejecting further measurements from the faulty sensor or signal. However, FDI requires parallel integrated navigation solutions, as described in Section 15.4.2.

15.3 Kalman Filter Measurement Innovations

The measurement innovations, $\delta\mathbf{z}_k^-$, of a Kalman filter, defined in Section 3.2, provide an indication of whether the measurements and state estimates are consis-

tent. Innovation filtering may be used to detect large discrepancies immediately, while innovation sequence monitoring enables smaller discrepancies to be detected over time. Both are described here, together with methods of recovering the Kalman filter estimates following fault detection.

The measurement residuals, δz_k^+ , may also be used for sequence monitoring and have a smaller covariance, making them more sensitive to errors. However, in a Kalman filter, they need to be calculated specially, while the innovations and their covariance are computed as part of normal operation.

For a true Kalman filter, the measurement innovation vector is

$$\delta z_k^- = z_k - H_k \hat{x}_k^- \quad (15.1)$$

while, for an EKF (Section 3.4.1), it is

$$\delta z_k^- = z_k - h(\hat{x}_k^-) \quad (15.2)$$

The covariance of the innovations, C_k^- , comprises the sum of the measurement noise covariance and the error covariance of the state estimates transformed into measurement space. Thus:

$$C_k^- = H_k P_k^- H_k^T + R_k \quad (15.3)$$

which is the denominator of the Kalman gain.

The normalized innovations are defined as

$$y_{k,j}^- = \frac{\delta z_{k,j}^-}{\sqrt{C_{k,j,j}^-}} \quad (15.4)$$

In an ideal Kalman filter, these have zero-mean unit-variance Gaussian distributions, and successive values are almost independent once the estimated states have converged with their true counterparts. However, time-correlated system or measurement noise, differences between the true and modeled system noise and measurement noise covariances, neglection of error sources, closed-loop feedback of state estimates to nonlinear systems, and use of an extended Kalman filter all cause departures from this [2]. Therefore, in a practical system, the statistics of the normalized innovations should always be measured before designing integrity monitoring algorithms that use them.

15.3.1 Innovation Filtering

Innovation filtering is also known as spike filtering, measurement gating, or prefiltering, noting that the latter term is also applied to measurement averaging (see Section 3.3.2). It compares the magnitude of each normalized measurement innovation, y^- , with a threshold and rejects the measurement for that iteration where the threshold is exceeded. It is applied prior to the computation of the Kalman gain

(see Section 3.2.2). Where a measurement is rejected, the corresponding rows of \mathbf{H} and rows and columns of \mathbf{R} must be excluded from the Kalman gain calculation (3.15), and error covariance update (3.17), as well as the state vector update (3.16). With a threshold of 3, 99.73 percent of genuine measurements are passed by the innovations filter, where the normalized innovations have a zero-mean unit-variance Gaussian distribution. However, if the threshold is set too low, the state estimates will be biased toward their initialization values.

Innovation filtering is applied to the normalized innovations rather than the raw innovations because the measurement innovations vary in size under normal Kalman filter operation. They are larger when the state uncertainties are larger, following initialization or a significant gap in the measurement stream, or when the measurement noise is larger. Figure 15.1 illustrates this.

Where a navigation sensor produces measurements in the form of position, velocity, or attitude fixes, all measurements from that sensor at a given iteration should be rejected when any component fails the innovations filter. For GNSS and other radio navigation systems supplying ranging measurements, innovation filtering should be applied independently to measurements from each satellite or transmitter. However, if a large number of measurements fails the innovations

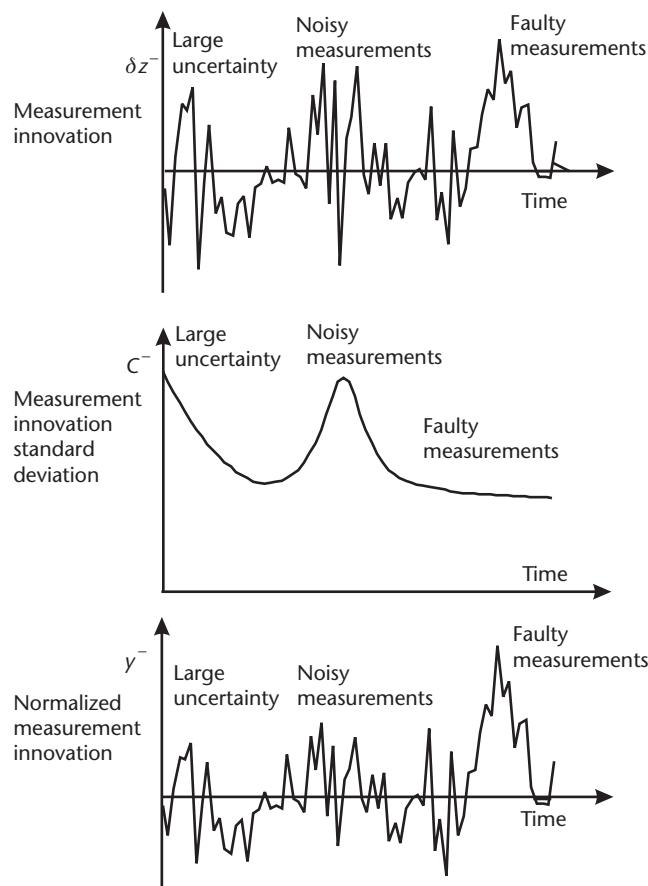


Figure 15.1 Raw and normalized measurement innovations.

filter simultaneously, the user equipment becomes suspect, so all measurements should be rejected. Where a GNSS pseudo-range measurement is rejected, the corresponding pseudo-range rate or ADR should also be rejected, but not vice versa.

Innovations filtering is a common feature of Kalman filter designs, even for applications with no formal integrity requirements. It is useful for filtering out short-term erroneous data, such as false fixes from feature-matching systems, magnetometer measurements affected by environmental magnetic anomalies, and measurements from GNSS tracking loops on the verge of loss of lock. It can also filter out spurious spikes in the measurement streams due to electrical interference, data communication errors, and timing problems. Such step changes in the measurements may be rejected without redundant measurement information where they are large enough.

Where a measurement repeatedly fails the innovations filter, it is indicative of a transient. However, this may be interpreted in a number of ways. A GNSS position solution may undergo a transient when a faulty signal is removed. Similarly, INS/GNSS velocity measurements used for transfer alignment can undergo a transient when GNSS is reacquired after an outage. A transient can also indicate that a fault has arisen. The source is sometimes ambiguous. A transient in an INS/GNSS integration filter affecting all measurements could be due to a fault in either the INS or the GNSS user equipment. Similarly, a transient in magnetometer measurements occurring just after initialization, when the confidence in previous measurements is low, or in feature-matching fixes, could indicate an error in either the new or the previous measurements. In general, the more information available, the easier it is to resolve the cause of a transient. Redundant information aids all forms of integrity monitoring.

Innovation filtering responds poorly to measurement errors that build up gradually. This is because the state estimates are contaminated by those errors before the fault becomes large enough to detect. Where there is a lack of redundant measurements, this keeps the measurement innovations small, so the fault is not detected at all. Otherwise, the contamination of the state estimates precludes effective fault isolation.

15.3.2 Innovation Sequence Monitoring

Smaller and slow-building discrepancies between the measurements and state estimates can be identified by forming test statistics from the last N measurements:

$$\mu_{kj} = \frac{1}{N} \sum_{i=k+1-N}^k \bar{y}_{i,j} \quad (15.5)$$

The standard deviation of the mean of N samples from a zero-mean unit-variance Gaussian distribution is $1/\sqrt{N}$. Therefore, a bias in the measurement innovations is identified where the following condition is met:

$$|\mu_{kj}| > \frac{T_{b\mu}}{\sqrt{N}} \quad (15.6)$$

where $T_{b\mu}$ is the innovation threshold. This is known as innovation sequence monitoring or innovation bias monitoring.

Where an innovation bias is detected, this may be due to a discrepancy between measurement streams or simply overoptimistic Kalman filter state uncertainties due to poor tuning. Overoptimistic state uncertainties will result in innovation biases on all measurement streams. A discrepancy between streams produces a much larger bias on the faulty measurement stream, but sometimes only where there are at least two redundant measurement streams.

A suitable innovation test statistic for the filter as a whole is [3]

$$s_{\delta z, k}^2 = \delta z_\mu^- {}^T C_\mu^- {}^{-1} \delta z_\mu^- \quad (15.7)$$

where

$$\begin{aligned} C_\mu^- {}^{-1} &= \sum_{i=k+1-N}^k C_i^- {}^{-1} \\ \delta z_\mu^- &= C_\mu^- \sum_{i=k+1-N}^k C_i^- {}^{-1} \delta z_i^- \end{aligned} \quad (15.8)$$

This has a chi-square distribution (see Section B.4 in Appendix B) with m degrees of freedom, where m is the number of components of the measurement vector, so comparing $s_{\delta z, k}$ against a threshold is sometimes known as a chi-square test.

Where a faulty measurement stream is identified, further measurements from that stream must be rejected. Depending on the cause of the error, the measurements may recover, so computation and monitoring of the measurement innovations may continue. Remedyng biased state estimates is discussed in Section 15.3.3. With a single Kalman filter, only FDR is available; FDI requires parallel filters as described in Section 15.4.2.

Selection of the sample size, N , is a tradeoff between response time and sensitivity, while selection of the threshold, $T_{b\mu}$, is a tradeoff between sensitivity and false-alarm rate. The faster the response time, the less contamination of the state estimates there will be. Multiple test statistics with different sample sizes and thresholds may be computed to enable larger biases to be detected more quickly [4]. Slowly increasing errors can sometimes be detected faster by applying an additional threshold to the rate of change of the test statistic as estimated by an additional Kalman filter [5].

One way of increasing the response time without increasing the false-alarm rate is to implement two thresholds with parallel filters [6]. When an innovation bias breaches the lower threshold, a parallel filter is spawned with the bias remedied, and further measurements from the suspect stream are rejected. If the upper threshold is then breached, the old filter is deleted, whereas if the upper threshold is not breached within a certain time, the new filter is deleted. Figure 15.2 illustrates this.

15.3.3 Remedyng Biased State Estimates

Where innovation biases due to overoptimistic Kalman filter state uncertainties have been detected, those uncertainties must be increased to give realistic values

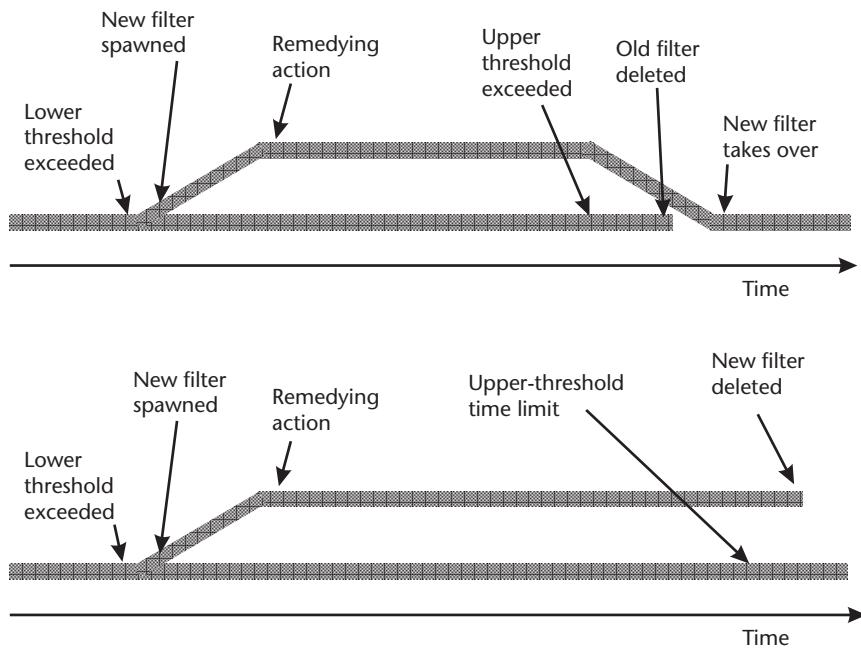


Figure 15.2 Applying parallel filters to innovation bias detection. (From: [7]. © 2002 QinetiQ Ltd. Reprinted with permission.)

and optimize the Kalman gain. Where the state estimates have been contaminated by erroneous measurement data or a transient that corrects a measurement stream has occurred, the Kalman filter must be made more receptive to new measurements in order to correct the estimates; this is also achieved by increasing the state uncertainties.

Increasing the state uncertainties is known as a *covariance reset* as the P matrix is reset or as a *Q boost/Q bump* because it effectively involves adding extra assumed system noise. Some reset techniques multiply the diagonal elements of the P matrix by a constant scaling factor; other techniques interpolate between the previous and initialization values. As the off-diagonal elements of P are initialized at zero, they may be too large when the diagonal elements are too small. Therefore many covariance reset techniques also scale down or zero the off-diagonal elements of P .

Following a covariance reset, the stored normalized residuals used for innovation sequence monitoring should be zeroed to prevent repeated triggering of the covariance reset. Alternatively, a relatively small covariance reset could be repeated over a series of Kalman filter iterations until innovation biases are no longer detected.

15.4 Direct Consistency Checks

Direct consistency checks compare quantities calculated from different combinations of measurements to determine whether they are consistent. Examples include

snapshot GNSS or terrestrial radio navigation solutions, IMU accelerometer and gyro measurements, and odometer or Doppler velocity measurements. They may also be used to compare complete navigation solutions, maintained in parallel using different combinations of measurements. Measurement consistency checks are described first, followed by integrity monitoring using parallel navigation solutions. GNSS measurement consistency checks are more commonly known as RAIM.

Where at least m measurements are required to compute the quantity under comparison, $m + 1$ measurements are needed for fault detection and $m + 2$ measurements for fault exclusion. For example, an IMU needs three accelerometers and three gyros to measure the specific force and angular rate in three dimensions. Therefore, four accelerometers and four gyros are needed for fault detection, and five of each sensor for fault exclusion [8]. Similarly, four pseudo-ranges are needed to determine a GNSS position solution, so five are needed for fault detection and six for fault exclusion. More measurements may be needed if the geometry of the signal line-of-sight vectors or instrument sensitive axes limits the observability of discrepancies. For example, consistency checks cannot be performed between inertial sensors with perpendicular sensitive axes, so skewed-sensor configurations tend to be used [8].

For measurement consistency checks, redundancy is required in a given type of measurement, while for parallel solutions, redundancy can be achieved across different sensor types. For example, GNSS measurements can be used to determine which of four accelerometers is faulty, enabling inertial navigation to continue using the other three.

15.4.1 Measurement Consistency Checks and RAIM

There are four main methods for performing measurement consistency checks in general or RAIM in particular [9]. The underlying principle is the same, so they all give similar performance.

Using the solution-separation method [10], if m measurements are available, then m different calculations of the position solution, specific force and angular rate, or velocity are made, each excluding a different measurement. Figure 15.3 illustrates this for GNSS RAIM. The differences between each calculation are then formed, and either the largest difference or the scalar average of all the differences is compared against a threshold. The test quantity exceeding the threshold denotes a fault.

The range comparison method [11] uses the first n measurements to calculate the position solution or other quantity under comparison. This is then used to predict the remaining $m - n$ measurements. If the measurements are consistent with the predictions, the consistency test is passed. Otherwise, a fault is declared.

The least-squares residual method first uses all the measurements, $\tilde{\mathbf{z}}$, to calculate a least-squares estimate of the n -component GNSS position and clock offset or other comparison quantity, \mathbf{x} , using

$$\hat{\mathbf{x}} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \tilde{\mathbf{z}} \quad (15.9)$$

where \mathbf{H} is the measurement matrix, analogous to that used in a Kalman filter (see Section 3.2.5). Measurement residuals are then computed using

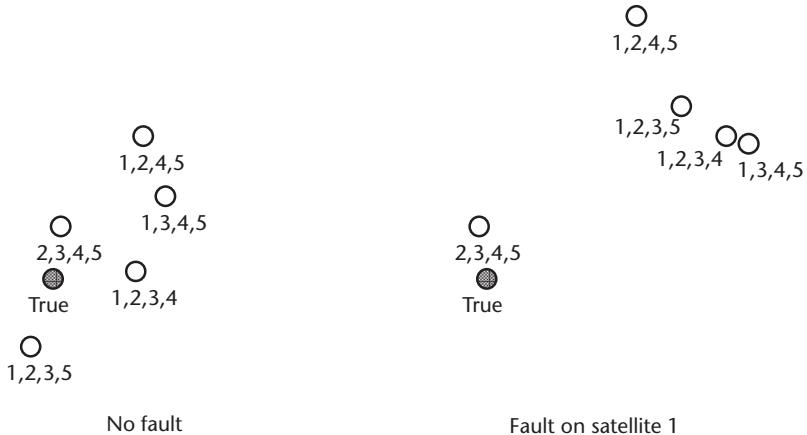


Figure 15.3 Separation of position solutions from different GNSS signal combinations.

$$\delta z^+ = \tilde{z} - H\hat{x} \quad (15.10)$$

Note that, for GNSS, the measurement, \tilde{z} , is the difference between the actual and predicted pseudo-ranges and \hat{x} is the difference between the estimated and predicted position and clock solutions, as described in Section 7.5.1. The largest or scalar-average residual may be compared against a threshold. However, it is more common to perform a chi-square test using $\delta z^{+T} \delta z^+$, or the normalized test statistic

$$s_{\delta z}^2 = \delta z^{+T} C^{+1} \delta z^+ \quad (15.11)$$

which has a chi-square distribution (see Section B.4 in Appendix B) with $m - n$ degrees of freedom, as the residuals are not independent [12]. C^+ is the measurement residual covariance matrix.

The parity method [13] uses the measurements, \tilde{z} , to compute an exact solution comprising the n component least-squares estimate of the comparison quantity, x , and an $m - n$ -component parity vector, p , the square of which, $p^T p$, is equal to $\delta z^{+T} \delta z^+$.

The thresholds for the test statistics are a tradeoff between sensitivity and false-alarm rate. For GNSS, terrestrial radio navigation, and IMU measurements, the test statistics may be averaged across successive iterations to improve both the sensitivity and false-alarm rate, making the measurement consistency checks analogous to Kalman filter innovation sequence monitoring (Section 15.3.2). Where carrier-smoothed pseudo-ranges are used, this averaging is implicit. However, time averaging can increase the response time. Odometer and Doppler velocity measurements exhibit transitory faults due to wheel slip or moving objects in the beam, so they are not suited to averaging.

When a fault has been detected, there are a number of ways of identifying the faulty measurement stream. With the least-squares residual and parity methods, the largest measurement residual, δz^+ , belongs to the faulty measurement, while with solution separation, the solution omitting the faulty measurement will have

a greater separation from the others. More generally, if there are at least two different measurements, the measurement consistency check may simply be repeated on m sets of $m - 1$ measurements. The set excluding the faulty measurement will pass the test and exhibit the lowest test statistic.

Where the measurements undergoing the consistency check are used exclusively to compute the final navigation solution, FDI is achieved simply by rejecting the faulty measurement. However, where the measurements are then input to a navigation or integration Kalman filter, or used for dead reckoning, only FDR is achievable, as undetected faulty measurement data may have been used earlier, particularly where time-averaged test statistics are used. In that case, parallel navigation solutions are needed for FDI.

15.4.2 Parallel Solutions

Parallel-solutions integrity monitoring maintains a number of parallel navigation solutions or filters, each excluding data from one sensor or radio navigation signal. Each additional navigation solution or Kalman filter is compared with the main one using a consistency test. If the test is failed, this indicates a fault in the sensor or signal omitted from one of the solutions. The navigation system output is then switched to the solution omitting the faulty sensor or signal. As this navigation solution has never incorporated data from the faulty source, isolation of the fault is achieved. Thus, the main benefit of parallel solutions is providing FDI. The main drawback is increased processor load.

Figure 15.4 shows how FDI may be applied to inertial navigation with redundant sensors by using multiple navigation processors. Each navigation solution, apart from the main one, omits one sensor. The processor load may be reduced by omitting accelerometer and gyro pairs. However, both sensors must then be rejected if a fault is detected.

Figure 15.5 illustrates parallel-solutions integrity monitoring for GNSS. Multiple Kalman filters are used, with each of the subfilters omitting data from one satellite.

Figure 15.6 shows the integrity monitoring architecture for INS/GNSS without redundant inertial sensors. Closed-loop corrections to the inertial navigation equations are fed back from the main integration filter only. However, they must also be fed back to the subfilters to enable them to correct their state estimates to account for the feedback to the INS. Where redundant inertial sensors are used, further subfilters based on inertial navigation solutions omitting individual sensors are added.

For multisensor integrated navigation in a centralized architecture (Section 14.1.3), the INS/GNSS integrity monitoring architecture is adapted to include subfilters omitting individual sensor or signal measurements or complete position or velocity solutions from a range of different types of navigation systems. In a federated architecture (Sections 14.1.4), parallel-solutions integrity monitoring may be implemented within the local filters, while measurement consistency checks, as described in Section 15.4.1, may be applied to the set of local filter outputs prior to fusing. Note that two local filters are needed for fault detection and three for fault isolation or recovery.

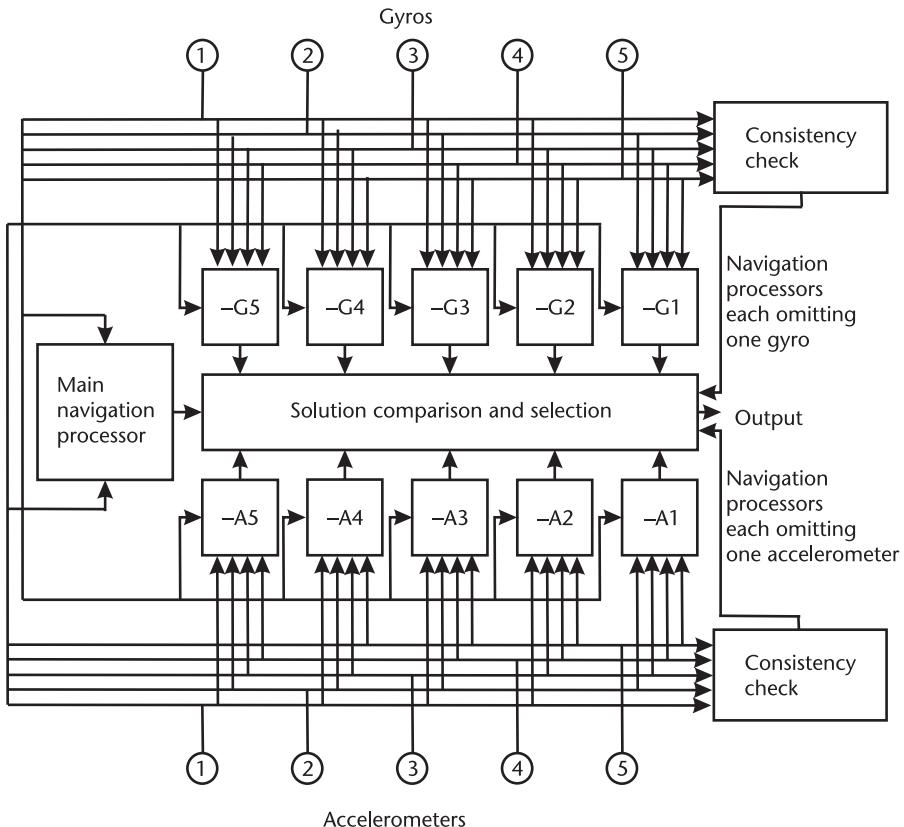


Figure 15.4 Parallel-solutions integrity monitoring applied to inertial navigation with redundant sensors.

Where parallel navigation solutions are used for stand-alone inertial navigation or dead reckoning, consistency checks may be performed at the measurement level, as described in Section 15.4.1, with the parallel solutions simply providing fault isolation.

For parallel Kalman filters, there are two ways of applying consistency checks. The extrapolation method [3] applies innovation sequence monitoring to each Kalman filter, as described in Section 15.3.2. The chi-square test statistic, (15.7), is usually computed. Where there is a fault, the test statistics for all filters except the one omitting the faulty data will exceed the threshold.

The solution-separation method [14] compares the state estimates of each subfilter, denoted by index j , with the main filter, denoted by index 0. A suitable chi-square test statistic is

$$s_{\delta x, k}^2 = (\hat{x}_{j, k}^+ - \hat{x}_{0, k}^+)^T \mathbf{B}_{j, k}^{+ -1} (\hat{x}_{j, k}^+ - \hat{x}_{0, k}^+) \quad (15.12)$$

where $\mathbf{B}_{j, k}^+ = E[(\hat{x}_{j, k}^+ - \hat{x}_{0, k}^+)(\hat{x}_{j, k}^+ - \hat{x}_{0, k}^+)^T]$ is the covariance of the state vector difference. From [4],

$$\mathbf{B}_{j, k}^+ = \mathbf{P}_{j, k}^+ - \mathbf{P}_{0, k}^+ \quad (15.13)$$

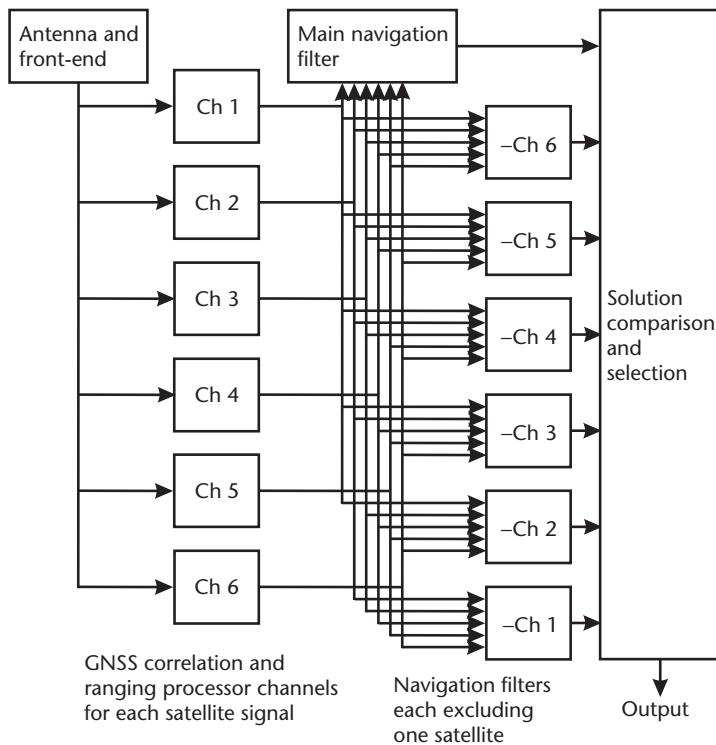


Figure 15.5 Parallel-solutions GNSS integrity monitoring.

noting that the state uncertainties of the main filter should be smaller than those of the subfilters.

Once a fault has been isolated, achieving fault exclusion requires the additional step of validating that the recovered navigation solution is fault free. It is more processor efficient to do this using innovation sequence monitoring, as using solution-separation tests would require a bank of subsubfilters, each excluding two measurement streams. Subsubfilters must be used where there is a requirement to isolate two faults.

15.5 Certified Integrity Monitoring

For safety-critical applications, such as civil aviation and many marine applications, it is not sufficient just to implement integrity monitoring; the navigation system must be certified to meet a guaranteed level of performance. This is known as the required navigation performance and is expressed in terms of accuracy, integrity, continuity, and availability.

The accuracy requirement simply states the average accuracy, across all conditions, that the navigation system must meet and is typically expressed as 95% bounds for the radial horizontal and vertical position errors (i.e., the errors that must not be exceeded more than 5 percent of the time). It is usually the easiest requirement to meet.

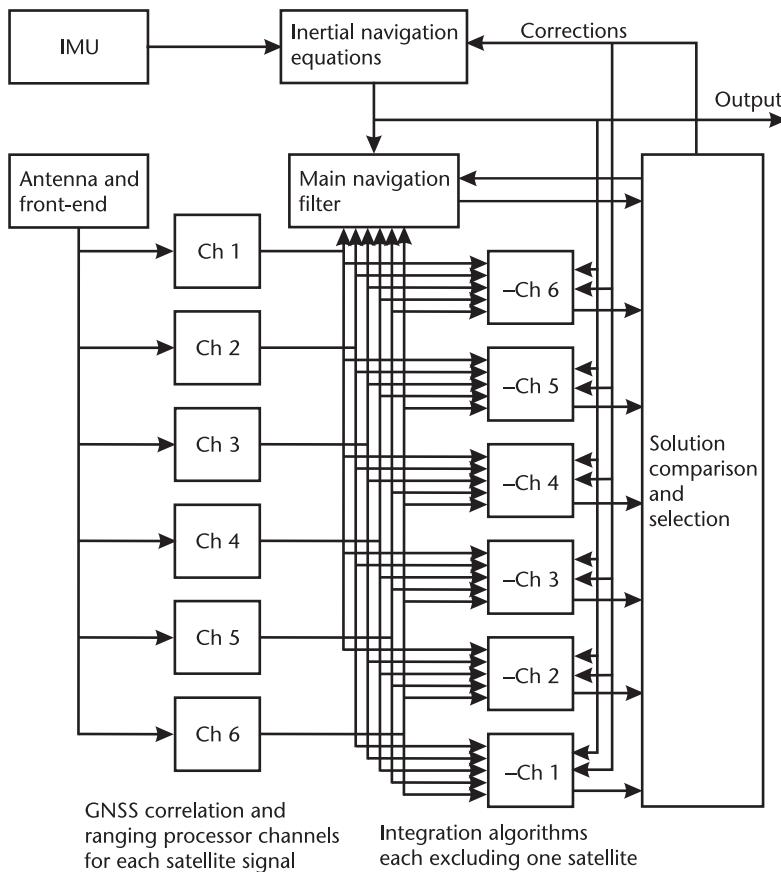


Figure 15.6 Parallel-filters INS/GNSS integrity monitoring (no redundant inertial sensors).

The integrity requirement is expressed in terms of a horizontal alert limit (HAL) and a vertical alert limit (VAL) that the radial horizontal and vertical position errors must not, respectively, exceed without alerting the user. Due to statistical outliers, absolute alert limits can never be enforced. Therefore a maximum probability of missed detection, p_{md} , must also be met. This is generally specified as a probability of integrity failure per unit time, which is the probability of a fault occurring in that time interval multiplied by the probability of missed detection. As a consequence, the fault detection tests described previously can give a third outcome in addition to fault detected and no fault detected. This is that the fault detection test is unavailable because it cannot guarantee to detect a fault that causes the HAL or VAL to be breached, in which case an integrity alert must be raised as if a fault had been detected.

The values compared in integrity tests are subject to uncertainty due to noise and unknown systematic errors. In a Kalman filter, these uncertainties are represented by the error covariance, P , and measurement noise covariance, R , matrices. Thus, the position errors at the fault detection threshold of an integrity test will have a probability distribution, as shown in Figure 15.7.

The integrity requirement allows the position errors exceeding the alert limit to remain undetected with a probability not exceeding p_{md} . A horizontal protection

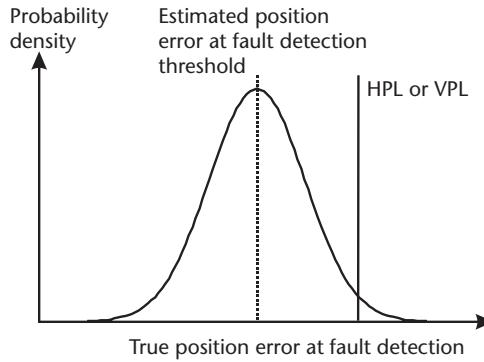


Figure 15.7 Position error distribution at fault detection threshold.

level (HPL) and vertical protection level (VPL) may then be defined such that the probability of the test statistic lying below the fault detection threshold is less than p_{md} for, respectively, radial horizontal and vertical position errors exceeding that level. Therefore, if the HPL exceeds the HAL or the VPL exceeds the VAL, an integrity alert is triggered.

In any integrity monitoring system, there is a tradeoff between the protection level, probability of missed detection, and false-alarm rate. For a given protection level, the detection threshold for the integrity-monitor test statistic determines the tradeoff between p_{md} and the false-alarm probability, p_{fa} , as Figure 15.8 illustrates [15].

For a test statistic, s , and threshold, T , p_{fa} and p_{md} are defined by

$$\int_T^{\infty} f_0(s) \, ds = p_{fa}, \quad \int_{-\infty}^T f_1(s) \, ds = p_{md} \quad (15.14)$$

where f_0 is the fault-free probability density function (see Section B.2 in Appendix B) of the test statistic, and f_1 is the PDF given a position error equal to the protection

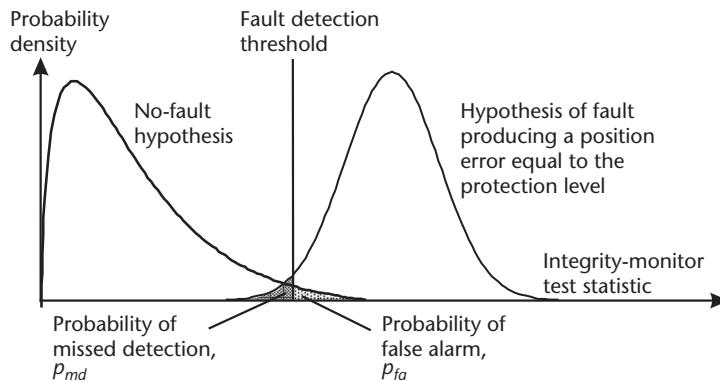


Figure 15.8 Missed-detection and false-alarm probability as a function of fault detection threshold.

level. To maintain a constant missed-detection probability and false-alarm rate, the detection threshold and protection level must vary. Where approximations are made in calculating the HPL and VPL, it is essential to ensure that the assumed error distribution overbounds its true counterpart to prevent computation of over-optimistic values of the HPL and VPL, which would constitute a safety risk [16].

The integrity requirements must be met for all faults; they are not averaged across the distribution of possible faults. Therefore, the HPL and VPL need only be calculated for the signal or sensor for which faults are most difficult to observe, as this produces the largest position error at the fault detection threshold. They are often expressed in terms of the largest value, across signals or sensors, of the ratio of the estimated position error to the test statistic, known as the *slope* [17, 18]. For GNSS and terrestrial radio navigation, the signal geometry is varying and not symmetric, so the slope varies between signals. By reweighting the least-squares position solution or Kalman filter measurements so that all signal faults are equally observable, the maximum slope is reduced, resulting in a smaller HPL and VPL. Integrity performance is thus improved, albeit at the expense of navigation-solution accuracy [19, 20].

Where FDE is implemented, an integrity alert is not automatically triggered when a fault is detected. It is triggered if the exclusion fails (i.e., the recovered navigation solution fails the fault detection test). If the exclusion is successful, the HPL and VPL are recalculated for the navigation solution excluding the faulty satellite. If the new HPL or VPL exceeds the HAL or VAL, the alert is triggered. Otherwise, the integrity test is passed and the navigation solution remains safe to use.

Essentially, the integrity monitoring system reports whether or not the HAL and VAL are protected, rather than whether or not a fault has occurred. This applies regardless of whether the HAL or VAL is breached due to a genuine fault, a false alarm, or insufficient data.

The final part of the integrity requirement is the time to alarm. This is the maximum time between the HAL or VAL being breached and the user being alerted of this.

Continuity is the ability to maintain a navigation solution within the HAL or VAL over a certain time window even if a fault occurs, noting that this requires FDE. This is important for applications such as instrument or automated landing of an aircraft, where there is a window of a certain length between the pilot committing to a landing and completing that landing. The pilot is unable to respond to integrity alerts during that window.

Continuity is determined by calculating the HPL and VPL of each navigation solution excluding one signal or sensor, propagated over the protection window. If any HPL or VPL breaches the HAL or VAL, a continuity alert is flagged to the user. The continuity requirement is specified in terms of the probability of a fault occurring within the protection window multiplied by the probability of the HAL or VAL being breached in the event of that fault.

The availability requirement is the proportion of time that the navigation system must operate with the accuracy requirement met and the integrity and continuity alert limits protected. Note that if the false-alarm rate is set high to minimize the HPL and VPL for a given missed-detection probability, the availability will be low.

Table 15.1 Example RNPs for Aircraft Navigation

<i>Operation</i>	<i>En Route (Oceanic)</i>	<i>Nonprecision Approach</i>	<i>Category I Landing</i>
Accuracy (95%)			
Horizontal	3.7 km	220m	16m
Vertical	—	—	—
Integrity failure rate	10^{-7} per hour	10^{-7} per hour	2×10^{-7} per approach
HAL	7.4 km	556m	40m
VAL	—	—	15–10m
Time to alert	5 minutes	10 seconds	6 seconds
Continuity failure rate	10^{-4} to 10^{-8} per hour	10^{-4} to 10^{-8} per hour	8×10^{-6} per 15 seconds
Availability	99–99.999 percent	99–99.999 percent	99–99.999 percent

For systems where the HPL and VPL vary, such as GNSS, the availability is calculated by determining the HPL and VPL across a grid of position and time, covering every possible signal geometry. The proportion of these HPLs and VPLs that fall within the HAL and VAL denotes the availability. This is easier for integrity monitoring methods where the HPL and VPL may be determined analytically. For innovation-based methods, the HPL and VPL calculations are empirical and must be validated using Monte Carlo simulations, which is not always practical [21].

Table 15.1 lists some example RNPs for aircraft navigation [15]. Stand-alone GPS fails to meet the availability and continuity requirements of practical RNPs due to an insufficient number of satellites. Less demanding RNPs, such as aircraft en-route navigation and nonprecision approach and marine harbor entrance and approach, can be met by GPS through integration with other navigation technologies, such as ELooran, use of SBAS signals, or addition of GLONASS or Galileo signals [22]. Meeting the RNP for aircraft landing additionally requires the use of differential GNSS.

References

- [1] Bhatti, U. I., and W. Y. Ochieng, “Failure Modes and Models for Integrated GPS/INS Systems,” *Journal of Navigation*, Vol. 60, No. 2, 2007, pp. 327–348.
- [2] Chaffee, J., K. Kovach, and G. Robel, “Integrity and the Myth of Optimal Filtering,” *Proc. ION NTM*, Santa Monica, CA, January 1997, pp. 453–461.
- [3] Diesel, J., and S. Luu, “GPS/IRS AIME: Calculation of Thresholds and Protection Radius Using Chi-Square Methods,” *Proc. ION GPS-95*, Palm Springs, CA, September 1995, pp. 1959–1964.
- [4] Young, R. S. Y., and G. A. McGraw, “Fault Detection and Exclusion Using Normalised Solution Separation and Residual Monitoring Methods,” *Navigation: JION*, Vol. 50, No. 3, 2003, pp. 151–169.
- [5] Bhatti, U. I., “An Improved Sensor Level Integrity Algorithm for GPS/INS Integrated System,” *Proc. ION GNSS2006*, Fort Worth, TX, September 2006, pp. 3012–3023.
- [6] Moore, S., G. Myers, and R. Hunt, “Future Integrated Navigation Guidance System,” *Proc. ION NTM*, Long Beach, CA, January 2001, pp. 447–457.
- [7] Groves, P. D., “Principles of Integrated Navigation,” Course Notes, QinetiQ Ltd., 2002.
- [8] Sukkarieh, S., et al., “A Low-Cost Redundant Inertial Measurement Unit for Unmanned Air Vehicles,” *International Journal of Robotics Research*, Vol. 19, No. 11, 2000, pp. 1089–1103.

- [9] Brown, R. G., "Receiver Autonomous Integrity Monitoring," in *Global Positioning System: Theory and Applications, Volume II*, B. W. Parkinson and J. J. Spilker, Jr., (eds.), Washington, D.C.: AIAA, 1996, pp. 143–165.
- [10] Brown, R. G., and P. W. McBurney, "Self-Contained GPS Integrity Check Using Maximum Solution Separation as the Test Statistic," *Proc. ION GPS-87*, Colorado Springs, CO, September 1987, pp. 263–268.
- [11] Lee, Y. C., "Analysis of Range and Position Comparison Methods as a Means to Provide GPS Integrity in the User Receiver," *Proc. ION 42nd AM*, Seattle, WA, June 1986, pp. 1–4.
- [12] Parkinson, B. W., and P. Axelrad, "Autonomous GPS Integrity Monitoring Using the Pseudorange Residual," *Navigation: JION*, Vol. 35, No. 2, 1988, pp. 255–274.
- [13] Sturza, M. A., "Navigation System Integrity Monitoring Using Redundant Measurements," *Navigation: JION*, Vol. 35, No. 4, 1988, pp. 483–501.
- [14] Brenner, M., "Integrated GPS/Inertial Fault Detection Availability," *Navigation: JION*, Vol. 43, No. 2, 1996, pp. 111–130.
- [15] Feng, S., et al., "A Measurement Domain Receiver Autonomous Integrity Monitoring Algorithm," *GPS Solutions*, Vol. 10, No. 2, 2006, pp. 85–96.
- [16] DeCleene, B., "Defining Pseudorange Integrity-Overbounding," *Proc. ION GPS 2000*, Salt Lake City, UT, September 2000, pp. 1916–1924.
- [17] Conley, R., et al., "Performance of Stand-Alone GPS," in *Understanding GPS Principles and Applications*, 2nd ed., E. D. Kaplan and C. J. Hegarty, (eds.), Norwood, MA: Artech House, 2006, pp. 301–378.
- [18] Powe, M., and J. Owen, "A Flexible RAIM Algorithm," *Proc. ION GPS-97*, Kansas, MO, September 1997, pp. 439–449.
- [19] Hwang, P. Y., and R. G. Brown, "RAIM FDE Revisited: A New Breakthrough in Availability Performance with NioRAIM (Novel Integrity-Optimized RAIM)," *Navigation: JION*, Vol. 53, No. 1, 2006, pp. 41–51.
- [20] Hwang, P. Y., "Applying NioRAIM to the Solution Separation Method for Inertially-Aided Aircraft Autonomous Integrity Monitoring," *Proc. ION NTM*, San Diego, CA, January 2005, pp. 992–1000.
- [21] Lee, Y. C., and D. G. O'Laughlin, "Performance Analysis of a Tightly Coupled GPS/Inertial System for Two Integrity Monitoring Methods," *Navigation: JION*, Vol. 47, No. 3, 2000, pp. 175–189.
- [22] Ochieng, W. Y., et al., "Potential Performance Levels of a Combined Galileo/GPS Navigation System," *Journal of Navigation*, Vol. 54, No. 2, 2001, pp. 185–197.

Vectors and Matrices

This appendix provides a refresher in vector and matrix algebra to support the main body of the book. Introductions to vectors and matrices are followed by descriptions of special matrix types, matrix inversion, and vector and matrix calculus [1–3].

A.1 Introduction to Vectors

A *vector* is a single-dimensional array of single-valued parameters, known as *scalars*. Here scalars are represented as italic and vectors as bold lower case. The scalar components of a vector are denoted by the corresponding italic symbol with a single numerical index and are normally represented together as a bracketed column. Thus,

$$\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \quad (\text{A.1})$$

where, in this case, the vector has n components or elements. Vectors may also be represented with an underline, \underline{a} , or an arrow, \vec{a} , while many authors do not limit them to lower case. Sometimes, it is convenient to represent a vector column on one line. Here, the notation $a = (a_1, a_2, \dots, a_n)$ is used. A vector is often used to represent a quantity that has both magnitude and direction; these vectors usually have three components. However, the components of a vector may also be unrelated, with different units. Both types of vector are used here.

Vectors are added and subtracted by adding and subtracting the components:

$$\mathbf{a} = \mathbf{b} + \mathbf{c} \Rightarrow \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} b_1 + c_1 \\ b_2 + c_2 \\ \vdots \\ b_n + c_n \end{pmatrix} \quad (\text{A.2})$$

The corresponding components must have the same units.

A vector is multiplied by a scalar simply by multiplying each component by that scalar:

$$\mathbf{a} = b\mathbf{c} \Rightarrow \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} bc_1 \\ bc_2 \\ \vdots \\ bc_n \end{pmatrix} \quad (\text{A.3})$$

Where two vectors have the same length, a scalar may be obtained by summing the products of the corresponding components. This is known as the *scalar product* or *dot product* and is written as

$$\mathbf{a} = \mathbf{b} \cdot \mathbf{c} = \sum_{i=1}^n b_i c_i \quad (\text{A.4})$$

Each component product, $b_i c_i$, must have the same units. Scalar products have the properties

$$\begin{aligned} \mathbf{a} \cdot \mathbf{b} &= \mathbf{b} \cdot \mathbf{a} \\ \mathbf{a} \cdot (\mathbf{b} + \mathbf{c}) &= \mathbf{a} \cdot \mathbf{b} + \mathbf{a} \cdot \mathbf{c} \end{aligned} \quad (\text{A.5})$$

Where the scalar product of two vectors is zero, they are said to be orthogonal.

Three-component vectors may also be combined to produce a three-component *vector product* or *cross product*:

$$\mathbf{a} = \mathbf{b} \wedge \mathbf{c} \Rightarrow \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} b_2 c_3 - b_3 c_2 \\ b_3 c_1 - b_1 c_3 \\ b_1 c_2 - b_2 c_1 \end{pmatrix} \quad (\text{A.6})$$

where all components within each vector must have the same units. The operator, \wedge , is often written as \times or \otimes . Vector products have properties

$$\begin{aligned} \mathbf{a} \wedge \mathbf{b} &= -\mathbf{b} \wedge \mathbf{a} \\ \mathbf{a} \wedge (\mathbf{b} \wedge \mathbf{c}) &= (\mathbf{a} \cdot \mathbf{c})\mathbf{b} - (\mathbf{a} \cdot \mathbf{b})\mathbf{c} \neq (\mathbf{a} \wedge \mathbf{b}) \wedge \mathbf{c} \\ \mathbf{a} \cdot (\mathbf{b} \wedge \mathbf{c}) &= \mathbf{b} \cdot (\mathbf{c} \wedge \mathbf{a}) = \mathbf{c} \cdot (\mathbf{a} \wedge \mathbf{b}) \\ \mathbf{a} \cdot (\mathbf{a} \wedge \mathbf{b}) &= \mathbf{b} \cdot (\mathbf{a} \wedge \mathbf{b}) = 0 \end{aligned} \quad (\text{A.7})$$

The magnitude of a vector is simply the square root of the scalar product of the vector with itself. Thus

$$a = |\mathbf{a}| = \sqrt{\mathbf{a} \cdot \mathbf{a}} \quad (\text{A.8})$$

A vector with magnitude 1 is known as a unit vector and is commonly denoted \mathbf{u} , \mathbf{e} , or $\mathbf{1}$. A unit vector may be obtained by dividing a vector by its magnitude. Thus,

$$\mathbf{u} = \mathbf{a}/a \Rightarrow \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix} = \begin{pmatrix} a_1/a \\ a_2/a \\ \vdots \\ a_n/a \end{pmatrix} \quad (\text{A.9})$$

A three-component unit vector can be used to represent a direction in three-dimensional space. The dot product of two unit vectors gives the angle between their directions:

$$\cos \theta_{ab} = \mathbf{u}_a \cdot \mathbf{u}_b \quad (\text{A.10})$$

The direction of a vector product is perpendicular to both the input vectors, while its magnitude is

$$|\mathbf{a} \wedge \mathbf{b}| = ab \sin \theta_{ab} \quad (\text{A.11})$$

Thus when \mathbf{a} and \mathbf{b} are parallel, their vector product is zero.

Finally, a vector may comprise an array of smaller vectors, known as subvectors, or a mixture of subvectors and scalars. For example,

$$\mathbf{a} = \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{pmatrix}, \quad \mathbf{a} = \begin{pmatrix} \mathbf{b} \\ \mathbf{c} \\ \mathbf{d} \end{pmatrix}, \quad \mathbf{a} = \begin{pmatrix} \mathbf{b} \\ c \\ d \end{pmatrix} \quad (\text{A.12})$$

A.2 Introduction to Matrices

A *matrix* is a two-dimensional array of scalars. It is represented as an upper case symbol, which here is also bold. The components are denoted by the corresponding italic letter with two indices, the first representing the row and the second the column. Thus,

$$\mathbf{A} = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mm} \end{pmatrix} \quad (\text{A.13})$$

where m is the number of rows, n is the number of columns, and mn is the number of elements. Where the components of a matrix do not have the same units, each row and column must have associated units with the units of each component being the product of the row and column units.

Matrices are added and subtracting by adding and subtracting the components:

$$\mathbf{A} = \mathbf{B} + \mathbf{C} \Rightarrow \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mn} \end{pmatrix} = \begin{pmatrix} B_{11} + C_{11} & B_{12} + C_{12} & \dots & B_{1n} + C_{1n} \\ B_{21} + C_{21} & B_{22} + C_{22} & \dots & B_{2n} + C_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ B_{m1} + C_{m1} & B_{m2} + C_{m2} & \dots & B_{mn} + C_{mn} \end{pmatrix} \quad (\text{A.14})$$

The corresponding components must have the same units.

Multiplication of two matrices produces the matrix of the scalar products of each row of the left matrix with each column of the right. Thus,

$$\mathbf{A} = \mathbf{BC} \Rightarrow \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{l1} & A_{l2} & \dots & A_{ln} \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^m B_{1j} C_{j1} & \sum_{j=1}^m B_{1j} C_{j2} & \dots & \sum_{j=1}^m B_{1j} C_{jn} \\ \sum_{j=1}^m B_{2j} C_{j1} & \sum_{j=1}^m B_{2j} C_{j2} & \dots & \sum_{j=1}^m B_{2j} C_{jn} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j=1}^m B_{lj} C_{j1} & \sum_{j=1}^m B_{lj} C_{j2} & \dots & \sum_{j=1}^m B_{lj} C_{jn} \end{pmatrix} \quad (\text{A.15})$$

where \mathbf{B} is an l -row by m -column matrix, \mathbf{C} is an $m \times n$ matrix, and \mathbf{A} is an $l \times n$ matrix. This operation may be described as premultiplication of \mathbf{C} by \mathbf{B} or postmultiplication of \mathbf{B} by \mathbf{C} . Matrices may only be multiplied where the number of columns of the left matrix matches the number of rows of the right matrix. Furthermore, each component product $B_{ij} C_{jk}$ must have the same units for a given i and k . A key feature of matrix multiplication is that reversing the order of the matrices produces a different result. In formal terms, matrices do not commute. Thus,

$$\mathbf{AB} \neq \mathbf{BA} \quad (\text{A.16})$$

Other matrix multiplication properties are

$$(\mathbf{AB})\mathbf{C} = \mathbf{A}(\mathbf{BC}) \quad (\text{A.17})$$

$$(\mathbf{A} + \mathbf{B})\mathbf{C} = \mathbf{AC} + \mathbf{BC}$$

A vector is simply a matrix with only one column. Therefore, the same rules apply for multiplying a vector by a matrix:

$$\mathbf{a} = \mathbf{B}\mathbf{c} \Rightarrow \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^n B_{1j}c_j \\ \sum_{j=1}^n B_{2j}c_j \\ \vdots \\ \sum_{j=1}^n B_{mj}c_j \end{pmatrix} \quad (\text{A.18})$$

where \mathbf{a} is an m -element vector, \mathbf{B} is an $m \times n$ matrix, and \mathbf{c} is an n -element vector.

The transpose of a matrix, denoted by the superscript T, reverses the rows and columns. Thus,

$$\mathbf{A}^T = \begin{pmatrix} A_{11} & A_{21} & \dots & A_{m1} \\ A_{12} & A_{22} & \dots & A_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ A_{1n} & A_{2n} & \dots & A_{mn} \end{pmatrix} \quad (\text{A.19})$$

If \mathbf{A} is an $m \times n$ matrix, then \mathbf{A}^T is an $n \times m$ matrix. The transpose of a matrix product has the property

$$(\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T \quad (\text{A.20})$$

The transpose of a vector is a single-row matrix:

$$\mathbf{a}^T = (a_1 \ a_2 \ \dots \ a_n) \quad (\text{A.21})$$

If a vector is premultiplied by the transpose of another vector, the result is the scalar product:

$$\mathbf{a}^T \mathbf{b} = \sum_{i=1}^n a_i b_i = \mathbf{a} \cdot \mathbf{b} \quad (\text{A.22})$$

This is known as the inner product. The outer product of two vectors, which may be of different sizes, is

$$\mathbf{ab}^T = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \dots & a_1 b_n \\ a_2 b_1 & a_2 b_2 & \dots & a_2 b_n \\ \vdots & \vdots & \ddots & \vdots \\ a_m b_1 & a_m b_2 & \dots & a_m b_n \end{pmatrix} \quad (\text{A.23})$$

where \mathbf{a} has m components and \mathbf{b} n .

Those matrix elements that have the same row and column index are known as diagonal elements. The remaining elements are off-diagonals. The sum of the diagonal elements is known as the trace, Tr . Thus,

$$\text{Tr}(\mathbf{A}) = \sum_{i=1}^{\min(m, n)} A_{ii} \quad (\text{A.24})$$

Matrices may also be made of smaller matrices, known as submatrices. For example,

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix} \quad (\text{A.25})$$

All submatrices in a given column must contain the same number of columns, while all submatrices in a given row must contain the same number of rows.

A.3 Special Matrix Types

Special types of matrix include the following:

- The square matrix, which has the same number of rows and columns;
- The zero matrix, $\mathbf{0}$, where all elements are zero;
- The identity matrix, \mathbf{I} , a square matrix that has unit diagonal and zero off-diagonal elements, such that $\mathbf{IA} = \mathbf{A}$;
- The diagonal matrix, which has zero off-diagonal elements;
- The symmetric matrix, which is a square matrix reflected about the diagonal such that $\mathbf{A}^T = \mathbf{A}$;
- The skew-symmetric, or antisymmetric, matrix, which has the property $\mathbf{A}^T = -\mathbf{A}$.

A 3×3 skew-symmetric matrix may be used to perform the vector product operation

$$\mathbf{Ab} = [\mathbf{a} \wedge] \mathbf{b} = \mathbf{a} \wedge \mathbf{b} \quad (\text{A.26})$$

where

$$\mathbf{A} = [\mathbf{a} \wedge] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \quad (\text{A.27})$$

An *orthonormal* or orthogonal matrix has the property

$$\mathbf{A}^T \mathbf{A} = \mathbf{A} \mathbf{A}^T = \mathbf{I} \quad (\text{A.28})$$

Furthermore, each row and each column forms a unit vector, and any pair of rows and any pair of columns are orthogonal. The product of two orthonormal matrices is also orthonormal as

$$\mathbf{AB}(\mathbf{AB})^T = \mathbf{AB}\mathbf{B}^T\mathbf{A}^T = \mathbf{A}\mathbf{I}\mathbf{A}^T = \mathbf{AA}^T = \mathbf{I} \quad (\text{A.29})$$

A.4 Matrix Inversion

The *inverse*, or reciprocal, of a matrix, \mathbf{A}^{-1} , fulfills the condition

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{AA}^{-1} = \mathbf{I} \quad (\text{A.30})$$

Thus, for an orthonormal matrix, the inverse and transpose are the same. Inversion takes the place of division for matrix algebra. Inverse matrices have the properties

$$\begin{aligned} (\mathbf{A}^{-1})^{-1} &= \mathbf{A} \\ (\mathbf{AB})^{-1} &= \mathbf{B}^{-1}\mathbf{A}^{-1} \\ \mathbf{A}^{-1}\mathbf{B} &\neq \mathbf{B}\mathbf{A}^{-1} \\ (\mathbf{A}^T)^{-1} &= (\mathbf{A}^{-1})^T \\ (b\mathbf{A})^{-1} &= \frac{1}{b}\mathbf{A}^{-1} \end{aligned} \quad (\text{A.31})$$

Not all matrices have an inverse. The matrix must be square and its rows (or columns) all linearly independent of each other. Where a matrix is not square, a pseudo-inverse, $\mathbf{A}^T(\mathbf{AA}^T)^{-1}$ or $(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$, may be used instead, provided \mathbf{AA}^T or $\mathbf{A}^T\mathbf{A}$ has an inverse.

The inverse of a matrix is given by

$$\mathbf{A}^{-1} = \frac{\text{adj}\mathbf{A}}{|\mathbf{A}|} \quad (\text{A.32})$$

where $\text{adj}\mathbf{A}$ and $|\mathbf{A}|$ are, respectively, the adjoint and determinant of \mathbf{A} . For an $m \times n$ matrix, these are given by

$$\text{adj}\mathbf{A} = \begin{pmatrix} \alpha_{11} & -\alpha_{12} & \dots & (-1)^{n+1}\alpha_{1n} \\ -\alpha_{21} & \alpha_{22} & \dots & (-1)^{n+2}\alpha_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ (-1)^{m+1}\alpha_{m1} & (-1)^{m+2}\alpha_{m2} & \dots & \alpha_{mn} \end{pmatrix} \quad (\text{A.33})$$

$$|\mathbf{A}| = \sum_{i=1}^n (-1)^{r+i} A_{ri} \alpha_{ri}$$

where r is an arbitrary row and α_{ri} is the minor, the determinant of \mathbf{A} excluding row r and column i . The solution proceeds iteratively, noting that the determinant of a 2×2 matrix is $A_{11}A_{22} - A_{21}A_{12}$. Alternatively, many numerical methods for matrix inversion are available.

A.5 Calculus

The derivative of a vector or matrix with respect to a scalar simply comprises the derivatives of the elements. Thus,

$$\frac{d\mathbf{a}}{db} = \begin{pmatrix} da_1/db \\ da_2/db \\ \vdots \\ da_n/db \end{pmatrix} \quad \frac{d\mathbf{A}}{db} = \begin{pmatrix} dA_{11}/db & dA_{12}/db & \dots & dA_{1n}/db \\ dA_{21}/db & dA_{22}/db & \dots & dA_{2n}/db \\ \vdots & \vdots & \ddots & \vdots \\ dA_{m1}/db & dA_{m2}/db & \dots & dA_{mn}/db \end{pmatrix} \quad (\text{A.34})$$

The derivative of a scalar with respect to a vector is written as the transposed vector of the partial derivatives with respect to each vector component. Thus,

$$\frac{da}{d\mathbf{b}} = \left(\frac{\partial a}{\partial b_1} \quad \frac{\partial a}{\partial b_2} \quad \dots \quad \frac{\partial a}{\partial b_n} \right) \quad (\text{A.35})$$

Postmultiplying this by the vector \mathbf{b} then produces a scalar with the same units as a .

The derivative of one vector with respect to another is then a matrix of the form

$$\frac{d\mathbf{a}}{d\mathbf{b}} = \begin{pmatrix} \partial a_1/\partial b_1 & \partial a_1/\partial b_2 & \dots & \partial a_1/\partial b_n \\ \partial a_2/\partial b_1 & \partial a_2/\partial b_2 & \dots & \partial a_2/\partial b_n \\ \vdots & \vdots & \ddots & \vdots \\ \partial a_m/\partial b_1 & \partial a_m/\partial b_2 & \dots & \partial a_m/\partial b_n \end{pmatrix} \quad (\text{A.36})$$

References

- [1] Farrell, J. A., and M. Barth, *The Global Positioning System and Inertial Navigation*, New York: McGraw-Hill, 1999.
- [2] Grewal, M. S., L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*, New York: Wiley, 2001.
- [3] Stephenson, G., *Mathematical Methods for Science Students*, 2nd ed., London, U.K.: Longman, 1973.

Statistical Measures

This appendix provides a brief refresher on statistical measures to support the main body of the book. The mean, variance, and standard deviation, probability density function, and the Gaussian and chi-square distributions are covered [1–3].

B.1 Mean, Variance, and Standard Deviation

The *mean*, μ_x , and *variance*, σ_x^2 , of a set of n observations, x_i , are

$$\mu_x = \frac{1}{n} \sum_{i=1}^n x_i, \quad \sigma_x^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \mu_x)^2 = \frac{1}{n} \sum_{i=1}^n x_i^2 - \mu_x^2 \quad (\text{B.1})$$

The *standard deviation*, σ_x , is simply the square root of the variance, while the root mean square (RMS) is

$$r_x = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \quad (\text{B.2})$$

Unbiased estimates of the mean, $\hat{\mu}$, and variance, $\hat{\sigma}^2$, of the distribution from which the observations are taken are

$$\hat{\mu} = \mu_x \quad \hat{\sigma}^2 = \frac{n}{n-1} \sigma_x^2 \quad (\text{B.3})$$

B.2 Probability Density Function

The *probability density function* (PDF), $f(x)$, of a distribution defines the relative probability that a sample, x , from that distribution takes a certain value such that

$$\int_{-\infty}^{\infty} f(x) dx = 1 \quad (\text{B.4})$$

The mean and variance of the distribution are defined by the expectations

$$\mu = E(x) = \int_{-\infty}^{\infty} xf(x) dx \quad (B.5)$$

$$\sigma^2 = E[(x - E(x))^2] = E(x^2) - E(x)^2 = \int_{-\infty}^{\infty} x^2 f(x) dx - \mu^2$$

The probability that x takes a value between x_- and x_+ is

$$p(x_- \leq x \leq x_+) = \int_{x_-}^{x_+} f(x) dx \quad (B.6)$$

This is the *confidence level*, p , that a random sample from a given distribution lies within the *confidence interval*, defined by the *confidence limits*, x_- and x_+ . Where the PDF is symmetric about the mean, the confidence limits are typically expressed as $x_{\pm} = \mu \pm n\sigma$, where the relationship between n and p depends on the type of distribution. Where $f(x)$ is zero for negative x , the lower confidence limit, x_- , is usually set to zero.

The *cumulative distribution function* is

$$F(X) = p(-\infty \leq x \leq X) = \int_{-\infty}^{X} f(x) dx \quad (B.7)$$

B.3 Gaussian Distribution

The *Gaussian* or *normal* distribution of a scalar, illustrated in Figure B.1, has PDF

$$f_G(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x - \mu)^2}{2\sigma^2}\right] \quad (B.8)$$

Samples from this distribution are denoted $N(\mu, \sigma)$. It is important for a number of reasons. Samples from a white noise process have a zero-mean Gaussian distribution. The central limit theorem states that the sum of independent random variables from an arbitrary distribution tends toward Gaussian as the number of samples increases. However, of most relevance here, a Kalman filter models all sources of random error with Gaussian distributions. Table B.1 gives confidence levels and limits for a zero mean Gaussian distribution.

The combined PDF of n variables with Gaussian distributions is that of the n -dimensional Gaussian distribution:

$$f_{G,n}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2} |\mathbf{P}|^{1/2}} \exp\left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \mathbf{P}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right] \quad (B.9)$$

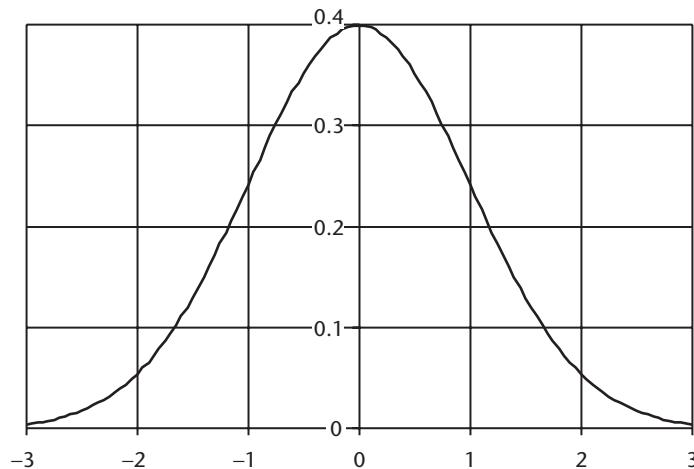


Figure B.1 PDF of a zero-mean unit-variance Gaussian distribution.

Table B.1 Confidence Levels and Symmetrical Limits for a Zero Mean Gaussian Distribution

Confidence Level	Symmetrical Confidence Limits (Standard Deviations, σ)
50%	± 0.675
68.2%	± 1
90%	± 1.645
95%	± 1.960
95.45%	± 2
99%	± 2.576
99.73%	± 3
99.9%	± 3.291
99.99%	± 3.891
99.9937%	± 4
99.999%	± 4.417

where the sample, \mathbf{x} , and mean $\boldsymbol{\mu}$, of the n variables are n -element vectors and the $n \times n$ covariance matrix is

$$\mathbf{P} = \mathbb{E}[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T] \quad (\text{B.10})$$

the diagonal elements of which are the variances of each component of \mathbf{x} .

B.4 Chi-Square Distribution

The chi-square statistic can be thought of as a measure of the degree to which a set of observations fit a normal distribution. It is defined as

$$\chi_n^2 = (\mathbf{x} - \boldsymbol{\mu})^T \mathbf{P}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \quad (\text{B.11})$$

where \mathbf{x} is the set of observations, $\boldsymbol{\mu}$ is their mean, and \mathbf{P} is their covariance. Where the observations, x_i , are independent, this simplifies to

$$\chi_n^2 = \sum_{i=1}^n \frac{(x_i - \mu_i)^2}{\sigma_i^2} \quad (\text{B.12})$$

The number of components of \mathbf{x} , n , is the number of degrees of freedom of the chi-square distribution. Its PDF is

$$f(\chi_n^2, n) = \frac{(1/2)^{n/2}}{\Gamma(n/2)} (\chi_n^2)^{(n/2-1)} \exp\left(-\frac{1}{2}\chi_n^2\right) \quad (\text{B.13})$$

where the gamma function, Γ , is

$$\Gamma(n/2) = \int_0^\infty x^{n/2-1} e^{-x} dx \quad (\text{B.14})$$

Figure B.2 shows the PDF and Table B.2 gives the confidence limits for chi-square distributions with 1 to 4 degrees of freedom.

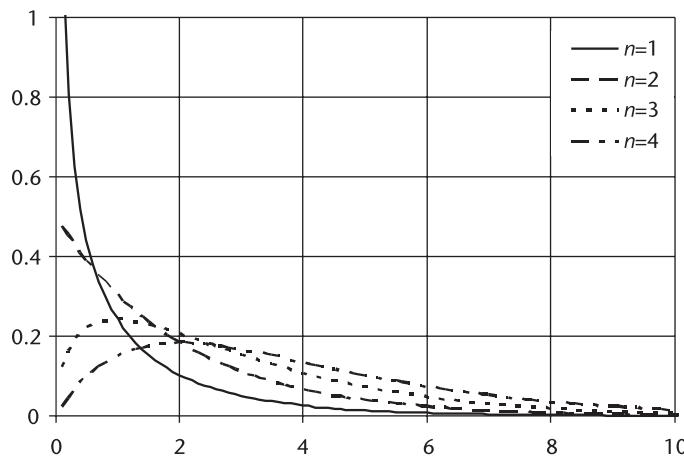


Figure B.2 PDF of chi-square distributions with 1 to 4 degrees of freedom.

Table B.2 Upper Confidence Limits (Lower Confidence Limit Is 0) for Chi-Square Distributions with N Degrees of Freedom

Confidence Level	Upper Confidence Limit			
	$n = 1$	$n = 2$	$n = 3$	$n = 4$
90 percent	2.71	4.61	6.25	7.78
95 percent	3.84	5.99	7.81	9.49
99 percent	6.64	9.21	11.34	13.28
99.9 percent	10.83	13.75	16.27	18.47

References

- [1] Boas, M. L., *Mathematical Methods in the Physical Sciences*, 2nd ed., New York: Wiley, 1983.
- [2] Brown, R. G., and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, 3rd ed., New York: Wiley, 1997.
- [3] Gelb, A., (ed.), *Applied Optimal Estimation*, Cambridge, MA: MIT Press, 1974.

List of Symbols

Here, the symbols that appear in the book's equations are listed. They are divided into matrices, denoted by upper case bold; vectors, denoted by lowercase bold; scalars, denoted by italics; subscripts and superscripts; and qualifiers. Subscripts and superscripts are only listed separately where they are used with more than one parent symbol; otherwise, the compound symbol is listed. Components of vectors and matrices are denoted by the equivalent scalar with subscript indices added. The magnitude of a vector is denoted by the equivalent scalar with no subscript index. Submatrices retain matrix notation but have subscript indices added.

Matrices

A	generic matrix
A	smoothing gain
B	generic matrix
B	covariance of the state vector difference
C	coordinate transformation matrix
C	generic matrix
C ⁻	covariance of measurement innovations
C ⁻	covariance of measurement residuals
F	system matrix
G	system noise distribution matrix
G	geometry matrix
G _g	gyro g-dependent errors
H	measurement matrix
I _n	$n \times n$ identity matrix (diagonal elements = 1, off-diagonal elements = 0)
J	measurement matrix for unestimated parameters
K	Kalman gain
M	scale factor and cross-coupling errors
P	error covariance matrix
P	distribution covariance matrix
Q	system noise covariance matrix
Q _U	system noise covariance matrix for unestimated parameters
R	measurement noise covariance matrix

T	position change transformation matrix
U	correlation matrix between states and unestimated parameters
W	error covariance matrix for unestimated parameters
Φ	transition matrix
Φ_U	transition matrix for unestimated parameters
Ψ	transition matrix linking states with unestimated parameters
Ω	skew-symmetric matrix of angular rate

Vectors

a	acceleration
a	generic vector
b	bias errors
b	generic vector
c	step length estimation coefficients
c	generic vector
c_i	<i>i</i> th row of coordinate transformation matrix
d	generic vector
f	specific force
f	system function
g	acceleration due to gravity
g	generic function
h	measurement function
h	angular momentum
k_n	Runge-Kutta integration intermediate step result
l	lever arm
m	cross-coupling errors
m	magnetic flux density
m	quantities measured
p	curvilinear position (geodetic latitude, longitude and geodetic height)
p	parity vector
q	quaternion attitude
r	Cartesian position
s	scale factor errors
s_{cg}	receiver clock g-dependent error coefficients
u	unit vector and line of sight unit vector
u	control vector
v	velocity
w	vector of white noise sources
w_m	measurement noise vector
w_s	system noise vector
x	generic vector or set of observations

\mathbf{x}	state vector
\mathbf{y}^-	normalized measurement innovation vector
\mathbf{y}^+	normalized measurement residual vector
\mathbf{z}	measurement vector
α	attitude increment
γ	acceleration due to the gravitational force
$\Delta \mathbf{r}$	position displacement
$\delta \mathbf{x}$	state vector residual
$\delta \mathbf{z}^-$	measurement innovation
$\delta \mathbf{z}^+$	measurement residual
η	flexure coefficients
μ	means
ρ	rotation vector
τ	torque
\mathbf{v}	integrated specific force
ψ	Euler attitude {roll, pitch, yaw} (no superscript)
ψ	Small-angle attitude (superscript indicates resolving axes)
ω	angular rate

Scalars

A	area
a	length of the semimajor axis
a	integer ambiguity
a	generic scalar
A_a	signal amplitude following amplification
a_f	satellite clock calibration coefficient
B	magnetic flux density
b	bias error
b	generic scalar
B_{L_CA}	carrier-phase tracking-loop bandwidth
B_{L_CF}	carrier-frequency tracking-loop bandwidth
B_{L_CO}	code tracking-loop bandwidth
B_{PC}	double-sided precorrelation bandwidth
C	spreading code
C	orbital harmonic correction term
c	speed of light in free space or fiber-optic coil
c	magnetic compass calibration coefficient
c	generic scalar
C/N_0	$10\log_{10}$ carrier power to noise density
c/n_0	carrier power to noise density

D	navigation data message
D	dilution of precision
D	code discriminator function
d	spacing of early and late correlation channels in code chips
d	depth
d	generic scalar
E	eccentric anomaly
e	eccentricity of the ellipsoid
e_o	eccentricity of the orbit
F	carrier-frequency discriminator function
F	cumulative probability
f	flattening of the ellipsoid
f	frequency
f	probability density function
f_a	ADC sampling frequency
H	orthometric height
h	geodetic height
h_i	mean ionosphere height
h	scaling factor in measurement matrix
I	intensity
i	inclination angle
J_2	Earth's second gravitational constant
k	discriminator gain
k_T	atmospheric temperature gradient
K	loop gain
L	geodetic latitude
l	number of system-noise-vector components
l	number of filter hypotheses
l	number of matrix rows
M	mean anomaly
M	narrowband to wideband accumulation interval
m	number of measurement-vector components
m	number of smoothing iterations
m	quantity measured
m	number of vector components or matrix rows/columns
N	geoid height
N	number of turns
N	normalization function
N	number of samples
N	sample from Gaussian distribution
n	root power spectral density
n	number of state-vector components

n	number of vector components or matrix columns
n	number of observations
n	number of degrees of freedom of chi-square distribution
n_k	number of measurement-vector hypotheses at iteration k
n_{rcd}	root power spectral density of receiver clock drift
n_0	noise power spectral density (not root)
P	power
p	first component of angular-rate vector
p	pressure
p	probability
q	second component of angular-rate vector
R_0	equatorial Earth radius
R	correlation function
R	gas constant
r	third component of angular-rate vector
r	iteration counter in summation
r	root mean square
R	average Earth radius
R_E	transverse radius of curvature
R_N	meridian radius of curvature
R_P	polar Earth radius
S	subcarrier function
s	signal amplitude
s	root chi-square test statistic
s	scale factor error
T	test statistic
T	temperature
T	track width
$T_{b\mu}$	innovation-bias threshold
t	time
t_{oe}	reference time of ephemeris
t_{sa}	time of signal arrival
t_{st}	time of signal transmission
t'_{st}	code phase
u	corrected argument of latitude
W	weighting factor
w	white noise source
x	generic process
x	first component of Cartesian position or a generic vector
x	code tracking error in code chips
x	generic argument of probability density
x_{-+}	confidence limits

x_- , x_+	confidence limits
y	second component of Cartesian position or a generic vector
z	third component of Cartesian position or a generic vector
Z_{cc}	correlator-comparison measurement
α	relative amplitude of multipath component
α	magnetic declination angle/variation
β	magnitude of the projection of position onto the equatorial plane
Γ	gamma function
γ	magnetic inclination/dip angle
Δ	range lag of multipath component
δ	range lag of multipath component in code chips
δ	Kronecker delta function (equals one when indices match and zero otherwise)
Δf	Doppler frequency shift
Δ_{ij}	scalar product of i th and j th coordinate transformation matrix rows
Δn	mean motion difference from computed value
Δr	distance traveled
Δx	rise time of signal waveform in code chips
$\Delta \rho_{dc}$	differential correction
$\delta \rho_e$	range error due to ephemeris data
$\Delta \rho_{ic}$	ionosphere correction
$\delta \rho_i$	ionosphere propagation error
$\delta \rho_{ie}$	Sagnac correction
$\delta \rho_m$	range error due to multipath
$\Delta \rho_R$	differenced pseudo-range
$\Delta \rho_{rc}$	relative receiver clock offset
$\delta \rho_{rc}$	receiver clock offset
$\Delta \rho_{r\phi}$	relative receiver phase offset
$\Delta \rho_{sc}$	satellite clock correction
$\delta \rho_s$	range error due to satellite clock error
$\Delta \rho_{tc}$	troposphere correction
$\delta \rho_t$	troposphere propagation error
$\delta \rho_w$	pseudo-range tracking error
$\delta \rho_\epsilon$	measurement residual of single-point navigation solution
θ	pitch or elevation angle
θ	generic angle
θ_{nu}	elevation angle of satellite line of sight vector
λ	longitude
Λ	likelihood
λ_{ca}	carrier wavelength
λ_0	wavelength
μ	resultant angle

μ	Earth's gravitational constant
μ	mean innovation test statistic
μ	mean
ν	true anomaly
ρ	range or pseudo-range
ρ	density
ρ_C	corrected pseudo-range measured by user equipment
ρ_R	pseudo-range measured by user equipment
ρ_T	true range
σ	standard deviation or error standard deviation
σ_{IQ}	noise standard deviation of accumulated correlator outputs
τ	correlation time
τ	propagation time
τ_a	correlator accumulation interval
τ_i	inertial navigation integration interval
τ_o	odometer measurement interval
τ_p	PDR measurement interval
τ_s	system propagation time
Φ	geocentric latitude
Φ	argument of latitude
Φ	carrier-phase discriminator function
ϕ	roll or bank angle
ϕ	phase
χ^2	chi-square statistic
ψ	yaw or heading angle
ψ_{bb}	boresight angle
ψ_{nu}	azimuth angle of satellite line of sight vector
Ω	longitude/right ascension of the ascending node
ω	angular frequency
ω	argument of perigee

Subscripts and Superscripts

A	denotes local magnetic anomalies
A	denotes accelerometer indicated
A	denotes attitude-matching transfer alignment measurement
a	denotes a vibrating element
a	denotes accelerometer
a	denotes user antenna body coordinate frame
a	denotes at the antenna
ASF	denotes additional secondary factor
B	denotes barometric height measurement

b	denotes body or INS body coordinate frame
b	denotes backward filter
b	denotes barometric altimeter
bad	denotes accelerometer dynamic bias
bgd	denotes gyro dynamic bias
C	denotes receiver-generated carrier
C	denotes postcorrelation
c	denotes from the coil
c	denotes due to or of coning motion
c	denotes cosine term
ca	denotes carrier or carrier phase
cf	denotes carrier frequency
co	denotes code
D	denotes down component
D	denotes Doppler measurement
D	denotes database-indicated
d	denotes at the detector
d	denotes dynamic
DC	denotes differentially corrected
E	denotes early correlation channel
E	denotes Earth's geomagnetic field
e	denotes Earth-centered Earth-fixed coordinate frame
F	denotes feature-matching measurement
f	denotes forward filter
f	denotes front-wheel coordinate frame
f	denotes fused solution
f	denotes feature-matching sensor body coordinate frame
G	denotes resultant position and time
G	denoted GNSS-derived
G	denotes Gaussian distribution
g	denotes gyro
GNSS	denotes GNSS partition
H	denotes horizontal
h	denotes height
h	denotes hard-iron
I	denotes in-phase
I	denotes ECI frame synchronized with ECEF at time of signal arrival
I	denotes INS-derived
i	generic index
i	denotes Earth-centered inertial coordinate frame
i	filter bank hypothesis index
i	denotes applicable to the inclination angle

<i>ic</i>	denotes ionosphere-corrected
<i>IF</i>	denotes intermediate frequency
<i>INS</i>	denotes INS partition
<i>j</i>	generic index
<i>j</i>	satellite or tracking channel number
<i>k</i>	iteration index for Kalman filter or tracking loop
<i>k</i>	generic index
<i>L</i>	denotes late correlation channel
<i>L</i>	denotes latitude
<i>L</i>	denotes left (wheel)
<i>L</i>	denotes leveling measurement
<i>M</i>	denotes magnetic heading measurement or error states
<i>m</i>	denotes Markov process
<i>m</i>	denotes pertaining to a multipath component with respect to the direct signal
<i>m</i>	denotes magnetometer-measured flux density and frame thereof
<i>m</i>	denotes magnetometer
<i>N</i>	denotes narrowband
<i>N</i>	denotes noise channel
<i>n</i>	denotes local navigation coordinate frame
<i>Nav</i>	denotes navigation solution
<i>ND</i>	denotes normalized code discriminator
<i>NED</i>	denotes nominal emission delay
<i>NF</i>	denotes normalized carrier-frequency discriminator
<i>NΦ</i>	denotes normalized carrier-phase discriminator
<i>O</i>	denotes odometer measurement
<i>o</i>	denotes orbital coordinate frame
<i>o</i>	denotes odometer
<i>P</i>	denotes position
<i>P</i>	denotes prompt correlation channel
<i>P</i>	denotes PDR measurement
<i>p</i>	denotes precession
<i>p</i>	denotes from the phase modulator
<i>PDR</i>	denotes PDR measurement
<i>Q</i>	denotes quadraphase
<i>Q</i>	denotes quasi-stationary alignment measurement
<i>R</i>	denotes right (wheel)
<i>R</i>	denotes terrestrial radio navigation and measurement thereof
<i>R</i>	denotes reference-navigation-system-indicated
<i>r</i>	denotes applicable to the orbit radius
<i>r</i>	denotes pseudo-range rate
<i>r</i>	denotes rear-wheel coordinate frame

r	denotes receiver
r	denotes random walk process
r	denotes reference body coordinate frame
ra	denotes accelerometer random noise
Ref	denotes reference navigation system
rg	denotes gyro random noise
S	denotes a point on the Earth's ellipsoidal surface
s	denotes a point on the Earth's geoid surface or water surface
s	denotes static
s	denotes due to or of sculling motion
s	denotes of the Schuler oscillation
s	denotes subcarrier
s	denotes sine term
s	denotes satellite body coordinate frame
s	denotes soft-iron
s	denotes scattering-surface coordinate frame
$Sensor$	denotes sensor
T	denotes the transpose of a matrix
T	denotes time
T	denotes TRN measurement
t	denotes due to tracking noise
t	denotes transmitter or transmitter body frame
t	denotes terrain
TD	denotes time difference
u	denotes applicable to the argument of latitude
V	denotes velocity-matching transfer alignment measurement
v	denotes oscillatory/vibratory
v	denotes velocity
VE	denotes very early correlation channel
VL	denotes very late correlation channel
W	denotes wideband
w	denotes wander-azimuth coordinate frame
w_lag	denotes lag-induced tracking error
x	denotes first component of a vector or axis
x	denotes cross-track component of velocity
y	denotes second component of a vector or axis
Z	denotes ZVU measurement
z	denotes third component of a vector or axis
α	denotes a generic object frame
α	generic index
β	denotes a generic reference frame
β	generic index

γ	denotes a generic set of resolving axes or frame
γ	generic index
δ	denotes a generic coordinate frame
ΔR	denotes radio navigation delta-range measurement
$\Delta\Delta$	denotes double-delta discriminator
Δo	denotes differential odometer
δx	denotes state vector residual
δz	denotes measurement innovation/residual
λ	denotes longitude
ρ	denotes range or pseudo-range
ρ_{rc}	denotes receiver clock offset
Σ	denotes a summation
ψ	denotes attitude/ GNSS attitude measurement
0	denotes value at the geoid
0	denotes initialization value
0	denotes a constant value
0	denotes at the reference time
0	denotes samples after carrier correlation and before code correlation
$-$	denotes after state propagation and before measurement update
$+$	denotes after measurement update
\perp	denotes perpendicular

See also the list of acronyms and abbreviations

Qualifiers

$E()$	expectation operator
(r)	denotes reference-station-indicated
(u)	denotes user-indicated
δ	denotes a small increment or error
Δ	denotes an increment, error or change
$'$	denotes alternative version
$"$	denotes alternative version
$^$	denotes estimate
\sim	denotes a navigation system measurement
$-$	denotes average value
$(-), -$	denotes at the beginning of the navigation processing cycle
$(+), +$	denotes at the end of the navigation processing cycle

List of Acronyms and Abbreviations

8PSK	8-phase shift key
ABAS	Aircraft-based augmentation system
ABS	Antilock braking system
ACU	Antenna control unit
ADC	Analog-to-digital converter
ADR	Accumulated delta range
AES	Aerospace and Electronic Systems
AFLT	Advanced forward-looking trilateration
AGARD	Advisory Group for Aerospace Research and Development
AGC	Automatic gain control
AGNSS	Assisted GNSS
AHRS	Attitude and heading reference system
AIAA	American Institute of Aeronautics and Astronautics
AltBOC	Alternate binary offset carrier
AM	Amplitude modulation
AM	Annual Meeting
ANFIS	Adaptive neuro-fuzzy inference system
ANN	Artificial neural network
AOA	Angle of arrival
AP	Access point
AR	Autoregressive
AS	Antispoofing
ASF	Additional secondary factor
ATAN	Arctangent
ATAN2	Arctangent (four quadrant)
ATC	Adaptive tightly coupled
ATF	Adaptive transversal filter
AtoN	Aid to navigation
Autonav	Automatic navigation
AUV	Autonomous underwater vehicle
baro	barometric altimeter
BOC	Binary offset code
BPF	Bandpass filter

BPSK	Biphase shift key
C/A	Coarse/acquisition
CASM	Coherent adaptive subcarrier modulation
CCD	Charge-coupled device
CDMA	Code-division multiple access
CE	Common era
CIRS	Conventional inertial reference system
CL	Civil long
CM	Civil moderate
COD	Cross over dot product
CP	Cross product
CRPA	Controlled-reception-pattern antenna
CS	Commercial service
CTP	Conventional terrestrial pole
CTRS	Conventional terrestrial reference system
CVL	Correlation velocity log
CVN	Continuous visual navigation
CZM	Conventional zero meridian
DCM	Direction cosine matrix
DCO	Digitally controlled oscillator
DDC	Decision-directed cross-product
DDM	Difference in depth of modulation
DDQ	Decision-directed quadrature
DEM	Digital elevation model
DGNSS	Differential GNSS
DGPS	Differential GPS
DLL	Delay lock loop
DME	Distance measuring equipment
DoD	Department of Defense
DOP	Dilution of precision
DPP	Dot-product power
DR	Dead reckoning
DTED	Digital Terrain Elevation Data
DTM	Digital terrain model
DVL	Doppler velocity log
ECEF	Earth-centered Earth-fixed
ECI	Earth-centered inertial
EGI	Embedded GNSS (or GPS) in INS
EGM 96	Earth Gravity Model 1996
EGNSS	Enhanced GNSS
EGNOS	European Geostationary Navigation Overlay System
EKF	Extended Kalman filter

ELE	Early-minus-late envelope
Eloran	Enhanced long-range navigation
ELP	Early-minus-late power
ENC	European Navigation Conference
EOTD	Enhanced-observed time difference
ESA	European Space Agency
EU	European Union
FDIS	Frequency-domain interference suppressor
FDMA	Frequency-division multiple access
FDE	Fault detection and exclusion
FDI	Fault detection and isolation
FDR	Fault detection and recovery
FEC	Forward error correction
FFR	Federated fusion-reset
FFT	Fast Fourier transform
FH	Frequency hopping
FLL	Frequency lock loop
FM	Frequency modulation
FMCW	Frequency-modulated continuous-wave
FNR	Federated no-reset
FOC	Full operational capability
FOG	Fiber-optic gyro
FTA	Fine time aiding
FWHM	Full width at half maximum
FZR	Federated zero-reset
GAGAN	GPS/GLONASS and geo-augmented navigation
GBAS	Ground-based augmentation system
GBCC	Ground-based control complex
GCS	Ground control segment
GDOP	Geometric dilution of precision
GLONASS	Global Navigation Satellite System or Global'naya Navigatsion-naya Sputnikovaya Sistema
GMS	Ground mission segment
GNSS	Global Navigation Satellite Systems
GPS	Global positioning system
GRAS	Ground-based regional augmentation system
GRI	Group repetition interval
GSA	GNSS Supervisory Authority
GSM	Global standard for mobile communications
GSS	Galileo sensor stations
GST	Galileo System Time
GTRF	Galileo terrestrial reference frame

gyro	Gyroscope
HAL	Horizontal alert limit
HARS	Heading and attitude reference system
HDOP	Horizontal dilution of precision
HOW	Handover word
HP	High performance
HPL	Horizontal protection level
HRG	Hemispherical resonator gyro
HZA	High zenith antenna
IAE	Innovation-based adaptive estimation
IAIN	International Association of Institutes of Navigation
IBLS	Integrity beacon landing system
ICD	Interface control document
ID	Identification
IEEE	Institute of Electrical and Electronic Engineers
IEE	Institute of Electrical Engineers
IERS	International Earth rotation and reference systems service
IF	Intermediate frequency
IFOG	Interferometric fiber-optic gyro
IGMAP	Iterative Gaussian mixture approximation of the posterior
IGRF	International Geomagnetic Reference Field
IGS	International GNSS service
ILS	Instrument landing system
IMU	Inertial measurement unit
INS	Inertial navigation system
INU	Inertial navigation unit
IOC	Initial operational capability
IODA	Issue of data almanac
IODC	Issue of data clock
IODE	Issue of data ephemeris
IODNav	Issue of data navigation
ION	Institute of Navigation
IOV	In-orbit validation
IPF	Integrity processing function
IQP	In-phase quadrature product
IR	Impulse radio
IRM	IERS reference meridian
IRNSS	Indian Regional Navigation Satellite System
IRP	IERS reference pole
IS	Interface standard
ITRF	International terrestrial reference frame
JION	<i>Journal of the Institute of Navigation</i>

JPO	Joint Program Office
JTIDS	Joint tactical information distribution system
KCPT	Kinematic carrier phase tracking
KF	Kalman filter
laser	Light amplification by stimulated emission and radiation
LAAS	Local area augmentation system
LADGNSS	Local area differential GNSS
L-AII	Legacy Accuracy Improvement Initiative
LAMBDA	Least-squares ambiguity decorrelation adjustment
LBL	Long baseline
LDC	Loran data channel
LHCP	Left-handed circular polarization
LEX	L-band experimental
LOP	Line of position
Loran	Long-range navigation
LSB	Least significant bit
L1/2/3/5	Link 1/2/3/5
L1C	Link 1 civil
L2C	Link 2 civil
L5I	Link 5 in-phase
L5Q	Link 5 quadrature
M	Military
MBOC	Modified binary offset carrier
MC	Multicarrier
MCMCMC	Metropolis-coupled Monte Carlo Markov chain
MCS	Master control station
MEDLL	Multipath-estimating delay lock loop
MEMS	Microelectromechanical systems
MEO	Medium Earth orbit
MET	Multipath elimination technology
MHKF	Multiple-hypothesis Kalman filter
MHT	Multiple-hypothesis tracking
MIDS	Multifunctional information distribution system
MIT	Massachusetts Institute of Technology
MLA	Multipath-limiting antenna
MMAE	Multiple-model adaptive estimation
MOEMS	Micro-optical-electromechanical systems
MOG	Micro-optic gyro
MSAS	MTSat satellite augmentation system
MTSat	Multifunction transport satellite
NASA	National Aeronautical and Space Administration
NATO	North Atlantic Treaty Organization

NAVSTAR	Navigation by Satellite Ranging and Timing
NCO	Numerically controlled oscillator
NDB	Nondirectional beacon
NED	Nominal emission delay
NGA	National Geospatial-Intelligence Agency
NIGCOMSAT	Nigerian communications satellite
NIMA	National Imagery and Mapping Agency
NTM	National Technical Meeting
OCS	Operational control segment
OCXO	Oven-controlled crystal oscillator
OS	Open service
P	Precise
PDAF	Probabilistic data association filter
PDF	Probability density function
PDOP	Position dilution of precision
PDR	Pedestrian dead reckoning
PIGA	Pendulous integrating gyro accelerometer
PLANS	Position, Location and Navigation Symposium
PLL	Phase lock loop
PLRS	Position location reporting source
ppm	Parts per million
PPP	Precise point positioning
PPS	Precise positioning system
PRN	Pseudo-random noise
PRS	Public regulated service
PSD	Power spectral density
PVT	Position, velocity and time
P(Y)	Precise (encrypted precise)
QC	Quadrature channel
QOI	Quadrature over in-phase
QPSK	Quadrature-phase shift key
QZSS	Quasi-Zenith Satellite System
radalt	Radar altimeter
RADGNSS	Regional area differential GNSS
RAIM	Receiver autonomous integrity monitoring
RelNav	Relative navigation
RF	Radio frequency
RFID	Radio-frequency identification
RFOG	Resonant fiber-optic gyro
RGNSS	Relative GNSS
RHCP	Right-handed circular polarization
RIN	Royal Institute of Navigation

RLG	Ring laser gyro
RMS	Root mean square
RNP	Required navigation performance
ROV	Remotely operated vehicle
RSS	Root sum of squares
RSS	Received signal strength
RTCM	Radio Technical Committee for Maritime Services
RTK	Real-time kinematic
RTS	Rauch, Tung, and Striebel
SA	Selective availability
SAR	Synthetic aperture radar
SAR	Search and rescue
SBAS	Space-based augmentation system
SC	Special Committee
SDR	Software-defined receiver
SF	Significant figures
SI	Système International d'unités
SITAN	Sandia Inertial Terrain Aided Navigation
SMAC	Scene matching by area correlation
SNAS	Satellite navigation augmentation system
SNU	Standard navigation unit
SOL	Safety of life
SPARTAN	Stockpot algorithm for robust terrain aided navigation
SPIE	Society of Photo-Optical Instrumentation Engineers
SPS	Standard positioning service
SV	Space vehicle
TACAN	Tactical air navigation
TAI	International Atomic Time
TAN	Terrain-aided navigation
TCAR	Three-carrier ambiguity resolution
TCM	Terrain contour matching
TCN	Terrain-contour navigation
TCXO	Temperature-compensated crystal oscillator
TD	Time difference
TDDM	Time-division data multiplexing
TDM	Time-division multiplex
TDMA	Time-division multiple access
TDOA	Time difference of arrival
TDOP	Time dilution of precision
TERCOM	Terrain Contour Matching
TERPROM	Terrain Profile Matching
TOA	Time of arrival

TOT	Time of transmission
TRN	Terrain-referenced navigation
TT&C	Telemetry, tracking, and control
UAV	Unmanned air vehicle
UDA	Undefined acronym
UDU	Upper diagonal upper
UERE	User-equivalent range error
UHF	Ultra-high frequency
U.K.	United Kingdom
UKF	Unscented Kalman filter
URA	User range accuracy
U.S.	United States
USNO	U.S. Naval Observatory
USSR	Union of Soviet Socialist Republics
UTC	Ultratightly coupled
UTC	Universal Coordinated Time
UWB	Ultra-wideband
VAL	Vertical alert limit
VBA	Vibrating-beam accelerometer
VDOP	Vertical dilution of precision
VHF	Very high frequency
VLBI	Very long baseline interferometry
VOR	VHF omnidirectional radiorange
VPL	Vertical protection level
WAAS	Wide Area Augmentation System
WADGNSS	Wide area differential GNSS
WCDMA	Wideband code division multiple access
WGS 84	World Geodetic System 1984
WLAN	Wireless local area network
WMM	World Magnetic Model
WSS	Wheel speed sensor
XO	Crystal oscillator
Y	Encrypted precise
ZUPT	Zero velocity update
ZVU	Zero velocity update

About the Author

Paul D. Groves holds a B.A. (with first class honors) in physics, an M.A. (Oxon), and a D.Phil. (Doctor of Philosophy) in experimental atomic and laser physics, all from the University of Oxford. He has been active in navigation systems research and development since joining the Defence Evaluation and Research Agency in January 1997. He transferred to QinetiQ Ltd. in July 2001 and, at the time of this writing, was a principal scientist in the Navigation and Positioning Algorithms team of the Autonomous Guidance and Telematics group.

Dr. Groves specializes in the integration and mathematical modeling of all types of navigation systems. His work has spanned aircraft, guided-weapon, pedestrian, and autonomous-underwater-vehicle navigation. He has developed algorithms for tightly coupled and deep INS/GNSS integration, multisensor integration, transfer alignment, quasi-stationary alignment, zero velocity updates, pedestrian dead reckoning, GNSS C/N₀ measurement, and inertial navigation. He has developed a high-fidelity GPS software simulation and contributed to the design of terrain-referenced navigation and visual navigation algorithms. He is an author of numerous conference and journal papers and has developed and presented the two-day course *Principles of Integrated Navigation*. He also holds a patent on adaptive tightly coupled integration.

Dr. Groves is a Fellow of the Royal Institute of Navigation and serves on its Technical and R&D Group Committees. He is also an active member of the Institute of Navigation and a chartered member of the Institute of Physics. He has helped to organize a number of conferences and seminars and acted as a peer reviewer for several journals.

