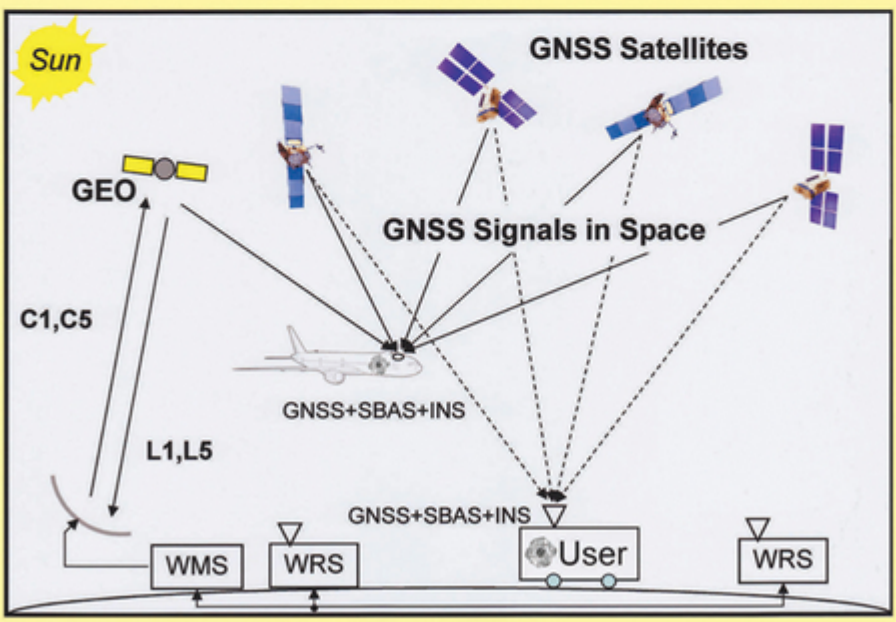
Third Edition

Global Navigation Satellite Systems, Inertial Navigation, and Integration

Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone



WILEY

GLOBAL NAVIGATION SATELLITE SYSTEMS, INERTIAL NAVIGATION, AND INTEGRATION

GLOBAL NAVIGATION SATELLITE SYSTEMS, INERTIAL NAVIGATION, AND INTEGRATION

THIRD EDITION

MOHINDER S. GREWAL
ANGUS P. ANDREWS
CHRIS G. BARTONE

A JOHN WILEY & SONS, INC., PUBLICATION

Published by John Wiley & Sons, Inc., Hoboken, New Jersey

Published simultaneously in Canada

Library of Congress Cataloging-in-Publication Data:

Grewal, Mohinder S.
Global navigation satellite systems, inertial navigation, and integration / Mohinder S. Grewal, Angus P. Andrews, Chris G. Bartone. – Third edition.
pages cm
Includes index.
Originally published under title: Global positioning systems, inertial navigation, and integration.
ISBN 978-1-118-44700-0 (cloth)
1. Global Positioning System. 2. Inertial navigation. 3. Kalman filtering. I. Andrews, Angus P. II. Bartone, Chris G. III. Title.
G109.5.G74 2013
910.285–dc23

2012032753

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

M.S.G. dedicates this book to the memory of his parents, Livlin Kaur and Sardar Sahib Sardar Karam Singh Grewal.
A.P.A. dedicates his contributions to his wife, Jeri, without whom it never would have happened.
C.G.B dedicates this work to his wife, Shirley, and two sons, Christopher and Stephen, for their never-ending support over the years.

CONTENTS

Preface xxvii

Acknowledgments xxxi

Acronyms and Abbreviations xxxiii

1 Introduction, 1

1.1 Navigation, 1
1.1.1 Navigation-Related Technologies, 1
1.1.2 Navigation Modes, 2
1.2 GNSS Overview, 4
1.2.1 GPS, 4
1.2.1.1 GPS Orbits, 4
1.2.1.2 GPS Signals, 4
1.2.1.3 Selective Availability (SA), 5
1.2.1.4 Modernization of GPS, 6
1.2.2 Global Orbiting Navigation Satellite System (GLONASS), 6
1.2.2.1 GLONASS Orbits, 6
1.2.2.2 GLONASS Signals, 6
1.2.2.3 Next Generation GLONASS, 7
1.2.3 Galileo, 7
1.2.3.1 Galileo Navigation Services, 7
1.2.3.2 Galileo Signal Characteristics, 8
1.2.3.3 Updates, 9

1.2.4 Compass (BeiDou-2), 10
1.2.4.1 Compass Satellites, 10
1.2.4.2 Frequency, 10
1.3 Inertial Navigation Overview, 10
1.3.1 Theoretical Foundations, 10
1.3.2 Inertial Sensor Technology, 11
1.3.2.1 Sensor Requirements, 12
1.3.2.2 Motivation, 13
1.3.2.3 Inertial Sensors Prior to Newton, 13
1.3.2.4 Early Momentum Wheel Gyroscopes (MWGs), 14
1.3.2.5 German Inertial Technology: 1930s–1945, 15
1.3.2.6 Charles Stark Draper (1901–1987), "The Father of Inertial Navigation", 19
1.3.2.7 Aerospace Inertial Technology, 20
1.3.2.8 Developments Since the Cold War, 30
1.4 GNSS/INS Integration Overview, 30
1.4.1 The Role of Kalman Filtering, 30
1.4.2 Implementation, 31
1.4.3 Applications, 31
1.4.3.1 Military Applications, 31
1.4.3.2 Civilian and Commercial Applications, 31
Problems, 32
References, 32

2 Fundamentals of Satellite Navigation Systems, 35

2.1 Navigation Systems Considered, 35
2.1.1 Systems Other than GNSS, 35
2.1.2 Comparison Criteria, 36
2.2 Satellite Navigation, 36
2.2.1 Satellite Orbits, 36
2.2.2 Navigation Solution (Two-Dimensional Example), 36
2.2.2.1 Symmetric Solution Using Two Transmitters on Land, 36
2.2.2.2 Navigation Solution Procedure, 40
2.2.3 Satellite Selection and Dilution of Precision (DOP), 41
2.2.4 Example Calculation of DOPS, 45
2.2.4.1 Four Satellites, 45
2.3 Time and GPS, 46
2.3.1 Coordinated Universal Time (UTC) Generation, 46
2.3.2 GPS System Time, 46
2.3.3 Receiver Computation of UTC, 47

2.4 Example: User Position Calculations with No Errors, 48
2.4.1 User Position Calculations, 48
2.4.1.1 Position Calculations, 48
2.4.2 User Velocity Calculations, 50
Problems, 51
References, 53

3 Fundamentals of Inertial Navigation, 54

3.1 Chapter Focus, 54
3.2 Basic Terminology, 55
3.3 Inertial Sensor Error Models, 59
3.3.1 Zero-Mean Random Errors, 60
3.3.1.1 White Sensor Noise, 60
3.3.1.2 Exponentially Correlated Noise, 60
3.3.1.3 Random Walk Sensor Errors, 60
3.3.1.4 Harmonic Noise, 61
3.3.1.5 "1/*f*" Noise, 61
3.3.2 Fixed-Pattern Errors, 61
3.3.3 Sensor Error Stability, 62
3.4 Sensor Calibration and Compensation, 63
3.4.1 Sensor Biases, Scale Factors, and Misalignments, 63
3.4.1.1 Compensation Model Parameters, 63
3.4.1.2 Calibrating Sensor Biases, Scale Factors, and Misalignments, 64
3.4.2 Other Calibration Parameters, 65
3.4.2.1 Nonlinearities, 65
3.4.2.2 Sensitivities to Other Measurable Conditions, 65
3.4.2.3 Other Accelerometer Models, 66
3.4.3 Calibration Parameter Instabilities, 66
3.4.3.1 Calibration Parameter Changes between Turn-Ons, 67
3.4.3.2 Calibration Parameter Drift, 67
3.4.4 Auxilliary Sensors before GNSS, 67
3.4.4.1 Attitude Sensors, 67
3.4.4.2 Altitude Sensors, 68
3.4.5 Sensor Performance Ranges, 68
3.5 Earth Models, 68
3.5.1 Terrestrial Navigation Coordinates, 69
3.5.2 Earth Rotation, 70
3.5.3 Gravity Models, 70
3.5.3.1 GNSS Gravity Models, 71
3.5.3.2 INS Gravity Models, 71
3.5.3.3 Longitude and Latitude Rates, 73

3.6 Hardware Implementations, 77
3.6.1 Gimbaled Implementations, 78
3.6.2 Floated Implementation, 80
3.6.3 Carouseling and Indexing, 81
3.6.3.1 Alpha Wander and Carouseling, 81
3.6.3.2 Indexing, 81
3.6.4 Strapdown Systems, 82
3.6.5 Strapdown Carouseling and Indexing, 82
3.7 Software Implementations, 83
3.7.1 Example in One Dimension, 83
3.7.2 Initialization in Nine Dimensions, 84
3.7.2.1 Navigation Initialization, 84
3.7.2.2 INS Alignment Methods, 84
3.7.2.3 Gyrocompass Alignment, 85
3.7.3 Gimbal Attitude Implementations, 87
3.7.3.1 Accelerometer Recalibration, 87
3.7.3.2 Vehicle Attitude Determination, 87
3.7.3.3 ISA Attitude Control, 88
3.7.4 Gimbaled Navigation Implementation, 89
3.7.5 Strapdown Attitude Implementations, 90
3.7.5.1 Strapdown Attitude Problems, 90
3.7.5.2 Coning Motion, 90
3.7.5.3 Rotation Vector Implementation, 93
3.7.5.4 Quaternion Implementation, 95
3.7.5.5 Direction Cosines Implementation, 96
3.7.5.6 MATLAB® Implementations, 97
3.7.6 Strapdown Navigation Implementation, 97
3.7.7 Navigation Computer and Software Requirements, 99
3.7.7.1 Physical and Operational Requirements, 100
3.7.7.2 Operating Systems, 100
3.7.7.3 Interface Requirements, 100
3.7.7.4 Software Development, 100
3.8 INS Performance Standards, 101
3.8.1 Free Inertial Operation, 101
3.8.2 INS Performance Metrics, 101
3.8.3 Performance Standards, 102
3.9 Testing and Evaluation, 102
3.9.1 Laboratory Testing, 102
3.9.2 Field Testing, 103
3.10 Summary, 103
Problems, 104
References, 106

4 GNSS Signal Structure, Characteristics, and Information Utilization, 108

4.1 Legacy GPS Signal Components, Purposes, and Properties, 109
4.1.1 Mathematical Signal Models for the Legacy GPS Signals, 109
4.1.2 Navigation Data Format, 112
4.1.2.1 Z-Count, 114
4.1.2.2 GPS Week Number (WN), 115
4.1.2.3 Information by Subframe, 116
4.1.3 GPS Satellite Position Calculations, 117
4.1.3.1 Ephemeris Data Reference Time Step and Transit Time Correction, 119
4.1.3.2 True, Eccentric, and Mean Anomaly, 119
4.1.3.3 Kepler's Equation for the Eccentric Anomaly, 120
4.1.3.4 Satellite Time Corrections, 121
4.1.4 C/A-Code and Its Properties, 122
4.1.4.1 Temporal Structure, 124
4.1.4.2 Autocorrelation Function, 124
4.1.4.3 Power Spectrum, 125
4.1.4.4 Despreading of the Signal Spectrum, 126
4.1.4.5 Role of Despreading in Interference Suppression, 127
4.1.4.6 Cross-Correlation Function, 128
4.1.5 P(Y)-Code and Its Properties, 129
4.1.5.1 P-Code Characteristics, 129
4.1.5.2 Y-Code, 130
4.1.6 L1 and L2 Carriers, 130
4.1.6.1 Dual-Frequency Operation, 130
4.1.7 Transmitted Power Levels, 131
4.1.8 Free Space and Other Loss Factors, 131
4.1.9 Received Signal Power, 132
4.2 Modernization of GPS, 132
4.2.1 Areas to Benefit from Modernization, 133
4.2.2 Elements of the Modernized GPS, 134
4.2.3 L2 Civil Signal (L2C), 135
4.2.4 L5 Signal, 136
4.2.5 M-Code, 138
4.2.6 L1C Signal, 139
4.2.7 GPS Satellite Blocks, 140
4.2.8 GPS III, 141
4.3 GLONASS Signal Structure and Characteristics, 141
4.3.1 Frequency Division Multiple Access (FDMA) Signals, 142

4.3.1.1 Carrier Components, 142
4.3.1.2 Spreading Codes and Modulation, 142
4.3.1.3 Navigation Data Format, 142
4.3.1.4 Satellite Families, 143
4.3.2 CDMA Modernization, 143
4.4 Galileo, 144
4.4.1 Constellation and Levels of Services, 144
4.4.2 Navigation Data and Signals, 144
4.5 Compass/BD, 146
4.6 QZSS, 146
Problems, 148
References, 150

5 GNSS Antenna Design and Analysis, 152

5.1 Applications, 152
5.2 GNSS Antenna Performance Characteristics, 152
5.2.1 Size and Cost, 153
5.2.2 Frequency and Bandwidth Coverage, 153
5.2.3 Radiation Pattern Characteristics, 155
5.2.4 Antenna Polarization and Axial Ratio, 156
5.2.5 Directivity, Efficiency, and Gain of a GNSS Antenna, 159
5.2.6 Antenna Impedance, Standing Wave Ratio, and Return Loss, 160
5.2.7 Antenna Bandwidth, 161
5.2.8 Antenna Noise Figure, 163
5.3 Computational Electromagnetic Models (CEMs) for GNSS Antenna Design, 164
5.4 GNSS Antenna Technologies, 166
5.4.1 Dipole-Based GNSS Antennas, 166
5.4.2 GNSS Patch Antennas, 166
5.4.2.1 Edge-Fed, LP, Single-Frequency GNSS Patch Antenna, 168
5.4.2.2 Probe-Fed, LP, Single-Frequency GNSS Patch Antenna, 170
5.4.2.3 Dual Probe-Fed, RHCP, Single-Frequency GNSS Patch Antenna, 171
5.4.2.4 Single Probe-Fed, RCHP, Single-Frequency GNSS Patch Antenna, 172
5.4.2.5 Dual Probe-Fed, RHCP, Multifrequency GNSS Patch Antenna, 175
5.4.3 Survey-Grade/Reference GNSS Antennas, 176
5.4.3.1 Choke Ring-Based GNSS Antennas, 176
5.4.3.2 Advanced Planner-Based GNSS Antennas, 177

5.5 Principles of Adaptable Phased-Array Antennas, 180
5.5.1 Digital Beamforming Adaptive Antenna Array Formulations, 182
5.5.2 STAP, 185
5.5.3 SFAP, 185
5.5.4 Configurations of Adaptable Phased-Array Antennas, 185
5.5.5 Relative Merits of Adaptable Phased-Array Antennas, 186
5.6 Application Calibration/Compensation Considerations, 187
Problems, 189
References, 190

6 GNSS Receiver Design and Analysis, 193

6.1 Receiver Design Choices, 193
6.1.1 Global Navigation Satellite System (GNSS) Application to be Supported, 193
6.1.2 Single or Multifrequency Support, 194
6.1.2.1 Dual-Frequency Ionosphere Correction, 194
6.1.2.2 Improved Carrier Phase Ambiguity Resolution in High-Accuracy Differential Positioning, 194
6.1.3 Number of Channels, 195
6.1.4 Code Selections, 195
6.1.5 Differential Capability, 196
6.1.5.1 Corrections Formats, 197
6.1.6 Aiding Inputs, 198
6.2 Receiver Architecture, 199
6.2.1 Radio Frequency (RF) Front End, 199
6.2.2 Frequency Down-Conversion and IF Amplification, 201
6.2.2.1 SNR, 202
6.2.3 Analog-to-Digital Conversion and Automatic Gain Control, 203
6.2.4 Baseband Signal Processing, 204
6.3 Signal Acquisition and Tracking, 204
6.3.1 Hypothesize about the User Location, 205
6.3.2 Hypothesize about Which GNSS Satellites Are Visible, 205
6.3.3 Signal Doppler Estimation, 206
6.3.4 Search for Signal in Frequency and Code Phase, 206
6.3.4.1 Sequential Searching in Code Delay, 208
6.3.4.2 Sequential Searching in Frequency, 209
6.3.4.3 Frequency Search Strategy, 209
6.3.4.4 Parallel and Hybrid Search Methods, 210

6.3.5 Signal Detection and Confirmation, 210
6.3.5.1 Detection Confirmation, 211
6.3.5.2 Coordination of Frequency Tuning and Code Chipping Rate, 213
6.3.6 Code Tracking Loop, 213
6.3.6.1 Code Loop Bandwidth Considerations, 217
6.3.6.2 Coherent versus Noncoherent Code Tracking, 217
6.3.7 Carrier Phase Tracking Loops, 218
6.3.7.1 PLL Capture Range, 221
6.3.7.2 PLL Order, 221
6.3.7.3 Use of Frequency-Lock Loops (FLLs) for Carrier Capture, 221
6.3.8 Bit Synchronization, 222
6.3.9 Data Bit Demodulation, 222
6.4 Extraction of Information for User Solution, 223
6.4.1 Signal Transmission Time Information, 223
6.4.2 Ephemeris Data for Satellite Position and Velocity, 224
6.4.3 Pseudorange Measurements Formulation Using Code Phase, 224
6.4.3.1 Pseudorange Positioning Equations, 226
6.4.4 Measurements Using Carrier Phase, 226
6.4.5 Carrier Doppler Measurement, 228
6.4.6 Integrated Doppler Measurements, 229
6.5 Theoretical Considerations in Pseudorange, Carrier Phase, and Frequency Estimations, 231
6.5.1 Theoretical Error Bounds for Code Phase Measurement, 232
6.5.2 Theoretical Error Bounds for Carrier Phase Measurements, 233
6.5.3 Theoretical Error Bounds for Frequency Measurement, 234
6.6 High-Sensitivity A-GPS Systems, 235
6.6.1 How Assisting Data Improves Receiver Performance, 236
6.6.1.1 Reduction of Frequency Uncertainty, 236
6.6.1.2 Determination of Accurate Time, 237
6.6.1.3 Transmission of Satellite Ephemeris Data, 238
6.6.1.4 Provision of Approximate Client Location, 238
6.6.1.5 Transmission of the Demodulated Navigation Bit Stream, 239
6.6.1.6 Server-Provided Location, 240

6.6.2 Factors Affecting High-Sensitivity Receivers, 240
6.6.2.1 Antenna and Low-Noise RF Design, 240
6.6.2.2 Degradation due to Signal Phase Variations, 240
6.6.2.3 Signal Processing Losses, 241
6.6.2.4 Multipath Fading, 241
6.6.2.5 Susceptibility to Interference and Strong Signals, 241
6.6.2.6 The Problem of Time Synchronization, 242
6.6.2.7 Difficulties in Reliable Sensitivity Assessment, 242
6.7 Software-Defined Radio (SDR) Approach, 242
6.8 Pseudolite Considerations, 243
Problems, 244
References, 246

7 GNSS Data Errors, 250

7.1 Data Errors, 250
7.2 Ionospheric Propagation Errors, 251
7.2.1 Ionospheric Delay Model, 252
7.2.2 GNSS SBAS Ionospheric Algorithms, 254
7.2.2.1 L1L2 Receiver and Satellite Bias and Ionospheric Delay Estimations for GPS, 256
7.2.2.2 Kalman Filter, 259
7.2.2.3 Selection of Q and R, 261
7.2.2.4 Calculation of Ionospheric Delay Using Pseudoranges, 262
7.3 Tropospheric Propagation Errors, 263
7.4 The Multipath Problem, 264
7.4.1 How Multipath Causes Ranging Errors, 264
7.5 Methods of Multipath Mitigation, 266
7.5.1 Spatial Processing Techniques, 267
7.5.1.1 Antenna Location Strategy, 267
7.5.1.2 Ground Plane Antennas, 267
7.5.1.3 Directive Antenna Arrays, 267
7.5.1.4 Long-Term Signal Observation, 267
7.5.2 Time-Domain Processing, 269
7.5.2.1 Narrow-Correlator Technology (1990–1993), 269
7.5.2.2 Leading-Edge Techniques, 270
7.5.2.3 Correlation Function Shape-Based Methods, 271
7.5.2.4 Modified Correlator Reference Waveforms, 271

7.5.3 Multipath Mitigation Technology (MMT) Technology, 272
7.5.3.1 Description, 272
7.5.3.2 Maximum-Likelihood (ML) Multipath Estimation, 272
7.5.3.3 The Two-Path ML Estimator (MLE), 273
7.5.3.4 Asymptotic Properties of ML Estimators, 274
7.5.3.5 The MMT Multipath Mitigation Algorithm, 274
7.5.3.6 The MMT Baseband Signal Model, 274
7.5.3.7 Baseband Signal Vectors, 275
7.5.3.8 The Log-Likelihood Function, 275
7.5.3.9 Secondary-Path Amplitude Constraint, 277
7.5.3.10 Signal Compression, 277
7.5.3.11 Properties of the Compressed Signal, 279
7.5.3.12 The Compression Theorem, 280
7.5.4 Performance of Time-Domain Methods, 281
7.5.4.1 Ranging with the C/A-Code, 281
7.5.4.2 Carrier Phase Ranging, 282
7.5.4.3 Testing Receiver Multipath Performance, 283
7.6 Theoretical Limits for Multipath Mitigation, 283
7.6.1 Estimation-Theoretic Methods, 283
7.6.1.1 Optimality Criteria, 284
7.6.2 Minimum Mean-Squared Error (MMSE) Estimator, 284
7.6.3 Multipath Modeling Errors, 284
7.7 Ephemeris Data Errors, 285
7.8 Onboard Clock Errors, 285
7.9 Receiver Clock Errors, 286
7.10 SA Errors, 288
7.11 Error Budgets, 288
Problems, 289
References, 291

8 Differential GNSS, 293

8.1 Introduction, 293
8.2 Descriptions of Local-Area Differential GNSS (LADGNSS), Wide-Area Differential GNSS (WADGNSS), and Space-Based Augmentation System (SBAS), 294
8.2.1 LADGNSS, 294
8.2.2 WADGNSS, 294
8.2.3 SBAS, 294
8.2.3.1 Wide-Area Augmentation System (WAAS), 294
8.2.3.2 European Global Navigation Overlay System (EGNOS), 298
8.2.3.3 Other SBAS, 299

8.3 GEO with L1L5 Signals, 299
8.3.1 GEO Uplink Subsystem Type 1 (GUST) Control Loop Overview, 302
8.3.1.1 Ionospheric Kalman Filters, 303
8.3.1.2 Range Kalman Filter, 303
8.3.1.3 Code Control Function, 304
8.3.1.4 Frequency Control Function, 304
8.3.1.5 L1L5 Bias Estimation Function, 305
8.3.1.6 L1L5 Bias Estimation Function, 305
8.3.1.7 Carrier Frequency Stability, 306
8.4 GUS Clock Steering Algorithm, 307
8.4.1 Receiver Clock Error Determination, 308
8.4.2 Clock Steering Control Law, 310
8.5 GEO Orbit Determination (OD), 310
8.5.1 OD Covariance Analysis, 312
8.6 Ground-Based Augmentation System (GBAS), 316
8.6.1 Local-Area Augmentation System (LAAS), 316
8.6.2 Joint Precision Approach and Landing System (JPALS), 317
8.6.3 Enhanced Long-Range Navigation (eLoran), 318
8.7 Measurement/Relative-Based DGNSS, 319
8.7.1 Code Differential Measurements, 319
8.7.1.1 Single-Difference Observations, 320
8.7.1.2 Double-Difference Observations, 320
8.7.2 Carrier Phase Differential Measurements, 321
8.7.2.1 Single-Difference Observations, 321
8.7.2.2 Double-Difference Observations, 321
8.7.2.3 Triple-Difference Observations, 322
8.7.2.4 Combinations of L1 and L2 Carrier Phase Observations, 322
8.7.3 Positioning Using Double-Difference Measurements, 322
8.7.3.1 Code-Based Positioning, 322
8.7.3.2 Carrier Phase-Based Positioning, 322
8.7.3.3 Real-Time Processing versus Postprocessing, 323
8.8 GNSS Precise Point Positioning Services and Products, 323
8.8.1 The International GNSS Service (IGS), 323
8.8.2 Continuously Operating Reference Stations (CORSs), 324
8.8.3 GPS Inferred Positioning System (GIPSY) and Orbit Analysis Simulation Software (OASIS), 324
8.8.4 Australia's Online GPS Processing System (AUPOS), 325
8.8.5 Scripps Coordinate Update Tool (SCOUT), 325
8.8.6 The Online Positioning User Service (OPUS), 325

Problems, 325
References, 326

9 GNSS and GEO Signal Integrity, 328

9.1 Introduction, 328
9.1.1 Range Comparison Method, 329
9.1.2 Least-Squares Method, 330
9.1.3 Parity Method, 331
9.2 SBAS and GBAS Integrity Design, 332
9.2.1 SBAS Error Sources and Integrity Threats, 333
9.2.2 GNSS-Associated Errors, 334
9.2.2.1 GNSS Clock Error, 334
9.2.2.2 GNSS Ephemeris Error, 335
9.2.2.3 GNSS Code and Carrier Incoherence, 335
9.2.2.4 GNSS Signal Distortion, 335
9.2.2.5 GNSS L1L2 Bias, 336
9.2.2.6 Environment Errors: Ionosphere, 336
9.2.2.7 Environment Errors: Troposphere, 336
9.2.3 GEO-Associated Errors, 336
9.2.3.1 GEO Code and Carrier Incoherence, 336
9.2.3.2 GEO-Associated Environment Errors: Ionosphere, 337
9.2.3.3 GEO-Associated Environment Errors: Troposphere, 337
9.2.4 Receiver and Measurement Processing Errors, 337
9.2.4.1 Receiver Measurement Error, 337
9.2.4.2 Intercard Bias, 337
9.2.4.3 Multipath, 338
9.2.4.4 L1L2 Bias, 338
9.2.4.5 Receiver Clock Error, 338
9.2.4.6 Measurement Processing Unpack/Pack Corruption, 338
9.2.5 Estimation Errors, 338
9.2.5.1 Reference Time Offset Estimation Error, 338
9.2.5.2 Clock Estimation Error, 339
9.2.5.3 Ephemeris Correction Error, 339
9.2.5.4 L1L2 Wide-Area Reference Equipment (WRE) and GPS Satellite Bias Estimation Error, 339
9.2.6 Integrity-Bound Associated Errors, 339
9.2.6.1 Ionospheric Modeling Errors, 339
9.2.6.2 Fringe Area Ephemeris Error, 340
9.2.6.3 Small-Sigma Errors, 340

9.2.6.4 Missed Message: Old But Active Data (OBAD), 340
9.2.6.5 Time to Alarm (TTA) Exceeded, 340
9.2.7 GEO Uplink Errors, 340
9.2.7.1 GEO Uplink System Fails to Receive SBAS Message, 340
9.2.8 Mitigation of Integrity Threats, 340
9.2.8.1 Mitigation of GNSS Associated Errors, 341
9.2.8.2 Mitigation of GEO-Associated Errors, 343
9.2.8.3 Mitigation of Receiver and Measurement Processing Errors, 343
9.2.8.4 Mitigation of Estimation Errors, 344
9.2.8.5 Mitigation of Integrity-Bound-Associated Errors, 345
9.3 SBAS Example, 346
9.4 Summary, 347
9.5 Future: GIC, 348
Problem, 348
References, 348

10 Kalman Filtering, 350

10.1 Introduction, 350
10.1.1 What Is a Kalman Filter?, 351
10.1.2 How Does It Work?, 352
10.1.2.1 Prediction and Correction, 353
10.1.3 How Is It Used?, 353
10.2 Kalman Filter Correction Update, 354
10.2.1 Deriving the Kalman Gain, 354
10.2.1.1 Approaches to Deriving the Kalman Gain, 355
10.2.1.2 Gaussian Probability Density Functions, 355
10.2.1.3 Properties of Likelihood Functions, 356
10.2.1.4 Solving for Combined Information Matrix, 358
10.2.1.5 Solving for Combined Argmax, 359
10.2.1.6 Noisy Measurement Likelihoods, 360
10.2.1.7 Gaussian Maximum-Likelihood Estimate (MLE), 362
10.2.1.8 Estimate Correction, 364
10.2.1.9 Kalman Gain Matrix for MLE, 364
10.2.2 Estimate Correction Using the Kalman Gain, 364
10.2.3 Covariance Correction for Using Measurements, 365
10.3 Kalman Filter Prediction Update, 365
10.3.1 Stochastic Systems in Continuous Time, 365
10.3.1.1 White-Noise Processes, 365
10.3.1.2 Stochastic Differential Equations, 365

10.3.1.3 Systems of First-Order Linear Differential Equations, 367
10.3.1.4 Representation in Terms of Vectors and Matrices, 368
10.3.1.5 Eigenvalues of Dynamic Coefficient Matrices, 369
10.3.1.6 Matrix Exponential Function, 371
10.3.1.7 Forward Solution, 371
10.3.1.8 Time-Invariant Systems, 371
10.3.2 Stochastic Systems in Discrete Time, 372
10.3.2.1 Zero-Mean White Gaussian Noise Sequences, 372
10.3.2.2 Gaussian Linear Stochastic Processes in Discrete Time, 372
10.3.3 State Space Models for Discrete Time, 373
10.3.4 Dynamic Disturbance Noise Distribution Matrices, 374
10.3.5 Predictor Equations, 374
10.4 Summary of Kalman Filter Equations, 375
10.4.1 Essential Equations, 375
10.4.2 Common Terminology, 375
10.4.3 Data Flow Diagrams, 376
10.5 Accommodating Time-Correlated Noise, 377
10.5.1 Correlated Noise Models, 378
10.5.1.1 Autocovariance Functions, 378
10.5.1.2 Random Walks, 378
10.5.1.3 Exponentially Correlated Noise, 379
10.5.1.4 Harmonic Noise, 379
10.5.1.5 Selective Availability (SA), 379
10.5.1.6 Slow Variables, 380
10.5.2 Empirical Modeling of Sensor Noise, 380
10.5.2.1 Spectral Characterization, 381
10.5.2.2 Shaping Filters, 381
10.5.3 State Vector Augmentation, 382
10.5.3.1 Correlated Dynamic Disturbance Noise, 382
10.5.3.2 Correlated Sensor Noise, 383
10.5.3.3 Correlated Noise in Continuous Time, 383
10.6 Nonlinear and Adaptive Implementations, 384
10.6.1 Assessing Linear Approximation Errors, 384
10.6.1.1 Statistical Measures of Acceptability, 384
10.6.1.2 Sampling for Acceptability Testing, 385
10.6.2 Nonlinear Dynamics, 390
10.6.2.1 Nonlinear Dynamics with Control, 390
10.6.2.2 Propagating Estimates, 390
10.6.2.3 Propagating Covariances, 390

10.6.3 Nonlinear Sensors, 391
10.6.3.1 Predicted Sensor Outputs, 391
10.6.3.2 Calculating Kalman Gains, 391
10.6.4 Linearized Kalman Filter, 391
10.6.5 Extended Kalman Filtering (EFK), 392
10.6.6 Adaptive Kalman Filtering, 393
10.7 Kalman–Bucy Filter, 395
10.7.1 Implementation Equations, 395
10.7.2 Kalman–Bucy Filter Parameters, 396
10.8 Host Vehicle Tracking Filters for GNSS, 397
10.8.1 Vehicle Tracking Filters, 397
10.8.2 Dynamic Dilution of Information, 397
10.8.2.1 Effect on Position Uncertainty, 398
10.8.3 Specialized Host Vehicle Tracking Filters, 399
10.8.3.1 Unknown Constant Tracking Model, 401
10.8.3.2 Damped Harmonic Resonator, 401
10.8.3.3 Type 2 Tracking Model, 402
10.8.3.4 DAMP1 Tracking Model: Velocity Damping, 403
10.8.3.5 DAMP2 Tracking Model: Velocity and Acceleration Damping, 403
10.8.3.6 DAMP3 Tracking Model: Position, Velocity, and Acceleration Damping, 405
10.8.3.7 Tracking Models for Highly Constrained Trajectories, 408
10.8.3.8 Filters for Spacecraft, 409
10.8.3.9 Other Specialized Vehicle Filter Models, 409
10.8.3.10 Filters for Different Host Vehicle Types, 409
10.8.3.11 Parameters for Vehicle Dynamics, 409
10.8.3.12 Empirical Modeling of Vehicle Dynamics, 409
10.8.4 Vehicle Tracking Filter Comparison, 411
10.8.4.1 Simulated Trajectory, 411
10.8.4.2 Results, 412
10.8.4.3 Model Dimension versus Model Constraints, 412
10.8.4.4 Role of Model Fidelity, 413
10.9 Alternative Implementations, 413
10.9.1 Schmidt–Kalman Suboptimal Filtering, 413
10.9.1.1 State Vector Partitioning, 414
10.9.1.2 Implementation Equations, 414
10.9.1.3 Simulated Performance in GNSS Position Estimation, 415
10.9.2 Serial Measurement Processing, 416
10.9.2.1 Measurement Decorrelation, 416
10.9.2.2 Serial Processing of Decorrelated Measurements, 417

10.9.3 Improving Numerical Stability, 417
10.9.3.1 Effects of Finite Precision, 417
10.9.3.2 Alternative Implementations, 418
10.9.3.3 Conditioning and Scaling Considerations, 419
10.9.4 Kalman Filter Monitoring, 421
10.9.4.1 Rejecting Anomalous Sensor Data, 421
10.9.4.2 Monitoring Filter Health, 423
10.10 Summary, 425
Problems, 426
References, 428

11 Inertial Navigation Error Analysis, 430

11.1 Chapter Focus, 430
11.2 Errors in the Navigation Solution, 432
11.2.1 The Nine Core INS Error Variables, 432
11.2.2 Coordinates Used for INS Error Analysis, 432
11.2.3 Model Variables and Parameters, 432
11.2.3.1 INS Orientation Variables and Errors, 433
11.2.4 Dynamic Coupling Mechanisms, 439
11.2.4.1 Dynamic Coupling, 439
11.3 Navigation Error Dynamics, 442
11.3.1 Error Dynamics due to Velocity Integration, 442
11.3.2 Error Dynamics due to Gravity Calculations, 443
11.3.2.1 INS Gravity Modeling, 443
11.3.2.2 Navigation Error Model for Gravity Calculations, 444
11.3.3 Error Dynamics due to Coriolis Acceleration, 445
11.3.4 Error Dynamics due to Centrifugal Acceleration, 446
11.3.5 Error Dynamics due to Earthrate Leveling, 447
11.3.6 Error Dynamics due to Velocity Leveling, 448
11.3.7 Error Dynamics due to Acceleration and Misalignments, 449
11.3.8 Composite Model from All Effects, 450
11.3.9 Vertical Navigation Instability, 452
11.3.9.1 Altimeter Aiding, 454
11.3.10 Schuler Oscillations, 457
11.3.11 Core Model Validation and Tuning, 459
11.3.11.1 Horizontal Inertial Navigation Model, 459
11.4 Inertial Sensor Noise, 459
11.4.1 CEP Rate versus Sensor Noise, 461
11.5 Sensor Compensation Errors, 461
11.5.1 Sensor Compensation Error Models, 462
11.5.1.1 Exponentially Correlated Parameter Drift Model, 463

11.5.1.2 Dynamic Coupling into Navigation Errors, 465
11.5.1.3 Augmented Dynamic Coefficient Matrix, 465
11.6 Software Sources, 467
11.7 Summary, 468
Problems, 470
References, 471

12 GNSS/INS Integration, 472

12.1 Chapter Focus, 472
12.1.1 Objective, 472
12.1.2 Order of Presentation, 473
12.2 GNSS/INS Integration Overview, 473
12.2.1 Historical Background, 473
12.2.2 The Loose/Tight Ranking, 475
12.2.2.1 Loosely Coupled Implementations, 476
12.2.2.2 More Tightly Coupled Implementations, 476
12.2.2.3 Ultratightly Coupled Integration, 477
12.2.2.4 Limitations, 477
12.2.3 Unified Navigation Model, 477
12.3 Unified Model for GNSS/INS Integration, 479
12.3.1 GNSS Error Models, 479
12.3.1.1 Receiver Clock Error Model, 479
12.3.1.2 Atmospheric Propagation Delay Model, 480
12.3.1.3 Pseudorange Measurement Noise, 481
12.3.2 INS Error Models, 481
12.3.2.1 Navigation Error Model, 481
12.3.2.2 Sensor Compensation Errors, 481
12.3.3 GNSS/INS Error Model, 482
12.3.3.1 State Variables, 482
12.3.3.2 Numbers of State Variables, 482
12.3.3.3 Dynamic Coefficient Matrix, 483
12.3.3.4 Process Noise Covariance, 484
12.3.3.5 Measurement Sensitivities, 484
12.4 Performance Analysis, 485
12.4.1 Dynamic Simulation Model, 485
12.4.1.1 State Transition Matrices (STMs), 485
12.4.1.2 Dynamic Simulation, 486
12.4.2 Results, 486
12.4.2.1 Stand-Alone GNSS Performance, 486
12.4.2.2 Stand-Alone INS Performance, 488
12.4.2.3 Integrated GNSS/INS Performance, 488
12.5 Other Integration Issues, 490
12.5.1 Antenna/ISA Offset Correction, 490
12.5.2 Influence of Trajectories on Performance, 491

12.6 Summary, 492
Problem, 493
References, 494

Appendix A Software, 495

A.1 Software Sources, 495
A.2 Software for Chapter 3, 496
A.3 Software for Chapter 4, 496
A.4 Software for Chapter 7, 496
A.5 Software for Chapter 10, 497
A.6 Software for Chapter 11, 498
A.7 Software for Chapter 12, 498
A.8 Almanac/Ephemeris Data Sources, 499

Appendix B Coordinate Systems and Transformations, 500

B.1 Coordinate Transformation Matrices, 500
B.1.1 Notation, 500
B.1.2 Definitions, 501
B.1.3 Unit Coordinate Vectors, 501
B.1.4 Direction Cosines, 502
B.1.5 Composition of Coordinate Transformations, 503
B.2 Inertial Reference Directions, 503
B.3 Application-Dependent Coordinate Systems, 504
B.3.1 Cartesian and Polar Coordinates, 504
B.3.2 Celestial Coordinates, 505
B.3.3 Satellite Orbit Coordinates, 505
B.3.4 ECI Coordinates, 507
B.3.5 Earth-Centered, Earth-Fixed (ECEF) Coordinates, 508
B.3.5.1 Longitudes in ECEF Coordinates, 508
B.3.5.2 Latitudes in ECEF Coordinates, 508
B.3.5.3 Latitude on an Ellipsoidal Earth, 509
B.3.5.4 Parametric Latitude, 509
B.3.5.5 Geodetic Latitude, 510
B.3.5.6 WGS84 Reference Geoid Parameters, 513
B.3.5.7 Geocentric Latitude, 513
B.3.5.8 Geocentric Radius, 514
B.3.6 Ellipsoidal Radius of Curvature, 515
B.3.7 Local Tangent Plane (LTP) Coordinates, 515
B.3.7.1 Alpha Wander Coordinates, 516
B.3.7.2 ENU/NED Coordinates, 516
B.3.7.3 ENU/ECEF Coordinates, 516
B.3.7.4 NED/ECEF Coordinates, 517
B.3.8 RPY Coordinates, 518

B.3.9 Vehicle Attitude Euler Angles, 518
B.3.9.1 RPY/ENU Coordinates, 519
B.3.10 GNSS Navigation Coordinates, 521
B.4 Coordinate Transformation Models, 523
B.4.1 Euler Angles, 523
B.4.2 Rotation Vectors, 524
B.4.2.1 Rotation Vector to Matrix, 525
B.4.2.2 Matrix to Rotation Vector, 527
B.4.2.3 Special Cases for $\sin(\theta) \approx 0$, 528
B.4.2.4 Time Derivatives of Rotation Vectors, 529
B.4.2.5 Time Derivatives of Matrix Expressions, 534
B.4.2.6 Partial Derivatives with Respect to Rotation Vectors, 537
B.4.3 Direction Cosine Matrix, 539
B.4.3.1 Rotating Coordinates, 540
B.4.4 Quaternions, 543
B.4.4.1 Quaternion Matrices, 543
B.4.4.2 Addition and Multiplication, 544
B.4.4.3 Conjugation, 545
B.4.4.4 Representing Rotations, 545
B.5 Newtonian Mechanics in Rotating Coordinates, 548
B.5.1 Rotating Coordinates, 548
B.5.2 Time Derivatives of Matrix Products, 549
B.5.3 Solving for Centrifugal and Coriolis Accelerations, 549

Index **551**

PREFACE

This book is intended for people who need a working knowledge of global navigation satellite systems (GNSSs), inertial navigation systems (INSs), and the Kalman filtering models and methods used in their integration. The book is designed to provide a usable, working familiarity with both the *theoretical* and *practical* aspects of these subjects. For that purpose, we include "real-world" problems from practice as illustrative examples. We also cover the more practical aspects of implementation: how to represent problems in a mathematical model, analyze performance as a function of model parameters, implement the mechanization equations in numerically stable algorithms, assess the computational requirements, test the validity of results, and monitor performance in operation with sensor data from Global Positioning System (GPS) and INS. These important attributes, often overlooked in theoretical treatments, are essential for effective application of theory to real-world problems.

The accompanying companion website (www.wiley.com/go/globalnavigation) contains MATLAB® m-files to demonstrate the workings of the navigation solutions involved. It includes Kalman filter algorithms with GNSS and INS data sets so that the reader can better discover how the Kalman filter works by observing it in action with GNSS and INS. The implementation of GNSS, INS, and Kalman filtering on computers also illuminates some of the practical considerations of finite-word-length arithmetic and the need for alternative algorithms to preserve the accuracy of the results. If the student wishes to apply what she or he learns, then it is essential that she or he experience its workings and failings—and learn to recognize the difference.

The book is organized for use as a text for an introductory course in GNSS technology at the senior level or as a first-year graduate-level course

in GNSS, INS, and Kalman filtering theory and applications. It could also be used for self-instruction or review by practicing engineers and scientists in these fields.

This third edition includes advances in GNSS/INS technology since the second edition in 2007, as well as many improvements suggested by reviewers and readers of the second edition. Changes in this third edition include the following:

1. Updates on the upgrades in existing GNSS systems and on other systems currently under development
2. Expanded coverage of basic principles of antenna design and practical antenna design solutions
3. Expanded coverage of basic principles of receiver design, and an update of the foundations for code and carrier acquisition and tracking within a GNSS receiver
4. Expanded coverage of inertial navigation, its history, its technology, and the mathematical models and methods used in its implementation
5. Derivations of dynamic models for the propagation of inertial navigation errors, including the effects of drifting sensor compensation parameters
6. Greatly expanded coverage of GNSS/INS integration, including derivation of a unified GNSS/INS integration model, its MATLAB implementations, and performance evaluation under simulated dynamic conditions

The accompanying website has also been augmented to include updated background material and additional MATLAB scripts for simulating GNSS-only and integrated GNSS/INS navigation. The CD-ROM includes satellite position determination, calculation of ionospheric delays, and dilution of precision.

Chapter 1 provides an overview of navigation, in general, and GNSS and inertial navigation, in particular. These overviews include fairly detailed descriptions of their respective histories, technologies, different implementation strategies, and applications.

Chapter 2 covers the fundamental attributes of satellite navigation systems, in general, and the technologies involved, how the navigation solution is implemented, and how satellite geometries influence errors in the solution.

Chapter 3 covers the fundamentals of inertial navigation, starting with its nomenclature and continuing through to practical implementation methods, error sources, performance attributes, and development strategies.

Chapters 4–9 cover basic theory of GNSS for a senior-level class in geomatics, electrical engineering, systems engineering, and computer science. Subjects covered in detail include basic GNSS satellite signal structures, practical receiver antenna designs, receiver implementation structures, error sources, signal processing methods for eliminating or reducing recognized error sources,

and system augmentation methods for improving system integrity and security.

Chapter 10 covers the fundamental aspects of Kalman filtering essential for GNSS/INS integration: its mathematical foundations and basic implementation methods, and its application to sensor integration in general and to GNSS navigation in particular. It also covers how the implementation includes its own performance evaluation and how this can be used in the predictive design of sensor systems.

Chapter 11 covers the basic errors sources and models for inertial navigation, including the effects of sensor noise and errors due to drifting inertial sensor error characteristics, how the resulting navigation errors evolve over time, and the resulting models that enable INS integration with other sensor systems.

Chapter 12 covers the essential mathematical foundations for GNSS/INS integration, including a unified navigation model, its implementation in MATLAB, evaluations of the resulting unified system performance under simulated dynamic conditions, and demonstration of the navigation performance improvement attainable through integrated navigation.

Appendix A contains brief descriptions of the MATLAB software on the CD-ROM, including formulas implementing the models developed in the different chapters and used for demonstrating how they work. Appendix B contains background material on coordinate systems and transformations implemented in the software, including derivations of the rotational dynamics used in navigation error modeling and GNSS/INS integration.

For instructors that wish to cover the fundamental aspects of GNSS, Chapters 1, 2, and 4–9 are recommended. Instructors of courses covering the fundamental concepts of inertial navigation can cover Chapters 1, 3, 10, and 11. A more advanced course in GNSS and INS integration should include Chapter 12, as well as significant utilization of the software routines provided for computer-based GNSS/INS integration projects.

Mohinder S. Grewal, Ph.D., P.E.
California State University at Fullerton

Angus P. Andrews, Ph.D.
Rockwell Science Center (retired)
Thousand Oaks, California

Chris G. Bartone, Ph.D., P.E.
Ohio University, Athens, Ohio

ACKNOWLEDGMENTS

We acknowledge Professors John Angus of Claremont Graduate University, Jay A. Farrell of the University of California, Riverside, and Richard B. Langley of the University of New Brunswick for assistance and inspiration on the outline of this edition. We acknowledge the assistance of Mrs. Laura A. Cheung of the Raytheon Company for her expert assistance in reviewing Chapter 8 ("Differential GNSS") and with the MATLAB® programs. Special thanks go to Dr. Larry Weill of California State University, Fullerton, for his contribution to Chapter 7 on multipath mitigation algorithms.

A. P. A. thanks Andrey Podkorytov at the Moscow Aviation Institute for corrections to the Schmidt–Kalman filter, Randall Corey from Northrop Grumman, and Michael Ash from C. S. Draper Laboratory for access to the developing Draft IEEE Standard for Inertial Sensor Technology; Dr. Michael Braasch at GPSoft, Inc. for providing evaluation copies of the GPSoft INS and GPS MATLAB® Toolboxes; Drs. Jeff Schmidt and Robert F. Nease, former Vice President of Engineering and Chief Scientist at Autonetics, respectively, for information on the early history of inertial navigation; Edward H. Martin for information on the very early history of GPS/INS integration; and Mrs Helen Boltinghouse for access to the personal memoirs of her late husband, Joseph C. Boltinghouse.

C. G. B. would like to thank Ohio University and many of its fine faculty, staff, and students with whom he has had the pleasure to interact in his research and teaching over the years. Such a rich environment has enabled him to develop a wide variety of classes and research efforts that these writings draw upon. Thanks also goes to Pat Fenton and Samantha Poon from NovAtel, Dave Brooks from Sensor Systems, James Horne from Roke, and Herbert Blaser from u-blox for providing antenna information.

ACRONYMS AND ABBREVIATIONS

3GPP	3rd Generation Partnership Project
A/D	Analog-to-digital (conversion)
ADC	Analog-to-digital converter
ADR	Accumulated delta range
ADS	Automatic dependent surveillance
AGC	Automatic gain control
A-GPS	Assisted GPS
AHRS	Attitude and heading reference system
AIC	Akaike information-theoretic criterion
AIRS	Advanced Inertial Reference Sphere
ALF	Atmospheric loss factor
ALS	Autonomous landing system
altBOC	Alternate binary offset carrier
AODE	Age of data word, ephemeris
AOR-E	Atlantic Ocean Region East (WAAS)
AOR-W	Atlantic Ocean Region West (WAAS)
APL	Applied Physics Laboratory, Johns Hopkins University
AR	Autoregressive
AR	Axial ratio
ARMA	Autoregressive moving average
ARP	Antenna reference point
ARNS	Aeronautical radio navigation services
ASD	Amplitude spectral density
ASIC	Application-specific integrated circuit
ASQF	Application-Specific Qualification Facility (EGNOS)
A-S	Antispoofing

ATC	Air traffic control
BD	BeiDou
BPF	Band pass filter
bps	bits per second
BOC	Binary offset carrier
BPSK	Binary phase-shift keying
BS	Base station
BW	Bandwidth
C	Civil
C/A	Coarse acquisition (channel or code)
C&V	Correction and verification (WAAS)
CDM	Code-division multiplexing
CDMA	Code-division multiple access
CEM	Computational electromagnetic model
CEP	Circular error probable, or circle of equal probability
CL	Code long
CM	Code moderate
CNAV	Civil navigation
CNMP	Code noise and multipath
CONUS	Conterminous United States, also Continental United States
CORS	Continuously operating reference station
COSPAS	Acronym from transliterated Russian title "Cosmicheskaya Sistyema Poiska Avariynich Sudov," meaning "Space System for the Search of Vessels in Distress"
CBOC	Combined BOC
C/NAV	Commercial navigation
CP	Carrier phase
cps	Chips per second
CRC	Cyclic redundancy check
CRPA	Controlled reception pattern antenna
CS	Control segment
CSDL	Charles Stark Draper Laboratory
CWAAS	Canadian WAAS
DC	Direct current
DDA	Digital differential analyzer
DGNSS	Differential GNSS
DGPS	Differential GPS
DLL	Delay-lock loop
DME	Distance measurement equipment
DOD	Department of Defense (United States)
DOP	Dilution of precision
DU	Delay unit
E	Eccentric anomaly
ECEF	Earth-centered, earth-fixed (coordinates)
ECI	Earth-centered inertial (coordinates)

EGNOS	European (also Geostationary) Navigation Overlay System
EIRP	Effective isotropic radiated power
EMA	Electromagnetic accelerator
EMI	Electromagnetic interference
ENU	East–north–up (coordinates)
ESA	European Space Agency
ESG	Electrostatic gyroscope
ESGN	Electrically Supported Gyro Navigator
EU	European Union
EWAN	EGNOS Wide-Area (communication) Network (EGNOS)
FAA	Federal Aviation Administration (United States)
FDMA	Frequency division multiple access
FEC	Forward error correction
FIR	Finite impulse response
FLL	Frequency-lock loop
FM	Frequency modulation
FOG	Fiber-optic gyroscope
FPE	Final prediction error (Akaike's)
FSLF	Free-space loss factor
F/NAV	Free navigation
ft	Feet
GAGAN	GPS & GEO Augmented Navigation (India)
GBAS	Ground-based augmentation system
GCCS	GEO Communication and Control Segment
GDOP	Geometric dilution of precision
GEO	Geostationary earth orbit
GES	GPS Earth Station COMSAT
GIC	GNSS Integrity Channel
GIOVE	Galileo In-Orbit Validation Element
GIPSY	GPS Infrared Positioning System
GIS	Geographic Information System(s)
GIVE	Grid ionosphere vertical error
GLONASS	Global Orbiting Navigation Satellite System
GNSS	Global navigation satellite system
GOA	GIPSY/OASIS analysis
GPS	Global Positioning System
GUS	GEO uplink subsystem
GUST	GEO uplink subsystem type 1
h	hour
HDOP	Horizontal dilution of precision
HEO	Highly-inclined elliptical orbit
HMI	Hazardously misleading information
HOW	Handover word
HPNS	High precision navigation signal (i.e., for GLONASS)
HRG	Hemispheric resonator gyroscope

I	In-phase
ICAO	International Civil Aviation Organization
ICC	Ionospheric correction computation
ICD	Interface control document
IDV	Independent Data Verification (of WAAS)
IF	Intermediate frequency
IFOG	Interferometric fiber-optic gyroscope
IGO	Inclined geosynchronous orbit
IGP	Ionospheric grid point (for WAAS)
IGS	International GNSS Service
ILS	Instrument landing system
IMU	Inertial measurement unit
Inmarsat	International Mobile (originally "Maritime") Satellite Organization
I/NAV	Integrity navigation
INS	Inertial navigation system
IOD	Issue of data
IODC	Issue of data clock
IODE	Issue of data ephemeris
IONO	Ionosphere, ionospheric
IOT	In-orbit test
IRU	Inertial reference unit
IS	Interface specification
ISA	Inertial sensor assembly
ITRF	International Terrestrial Reference Frame
ITU	International Telecommunications Union
JPALS	Joint precision approach and landing system
JTIDS	Joint Tactical Information Distribution System
km	Kilometer
KPA	Klystron Power Amplifier
LAAS	Local-Area Augmentation System
LADGPS	Local-area differential GPS
LAMBDA	Least-squares ambiguity decorrelation adjustment
LD	Location determination
LEM	Lunar Excursion Module
LHCP	Left-hand circularly polarized
LINCS	Local Information Network Communication System
LIS	Land Information Systems
LNA	Low-noise amplifier
LORAN	Long-range navigation
LOS	Line of sight
LP	Linear polarization
LPV	Lateral positioning with vertical guidance
LSB	Least significant bit
LTP	Local tangent plane

M	Mean anomaly
m	Meter
M	Military
MBOC	Modified BOC
MCC	Mission/Master Control Center (EGNOS)
Mcps	million chips per second
MEDLL	Multipath-estimating delay-lock loop
MEMS	Microelectromechanical system(s)
MEO	Medium earth orbit
MS	Mobile station (i.e., cell phone)
ML	Maximum likelihood
MLE	Maximum-likelihood estimation (or estimator)
MMIA	Mueller Mechanical Integrating Accelerometer
MMSE	Minimum mean-squared error (estimator)
MMT	Multipath mitigation technology
MOPS	Minimum Operational Performance Standards
MSAS	MTSAT Satellite-based Augmentation System (Japan)
MSB	Most significant bit
MTSAT	Multifunctional Transport Satellite (Japan)
MVDR	Minimum variance distortionless response
MVUE	Minimum-variance unbiased estimator
MWG	Momentum wheel gyroscope
NAS	National Airspace System
NAVSTAR	Navigation system with time and ranging
NCO	Numerically controlled oscillator
NED	North–east–down (coordinates)
NF	Noise figure
NGS	National Geodetic Survey (United States)
NLES	Navigation Land Earth Station(s) (EGNOS)
NOAA	National Oceanic and Atmospheric Administration
NPA	Nonprecision approach
NSRS	National Spatial Reference System
NSTB	National Satellite Test Bed
OASIS	Orbit analysis simulation software
OBAD	Old but active data
OD	Orbit determination
OPUS	Online Positioning User Service (of NGS)
OS	Open Service (of Galileo)
PA	Precision approach
PACF	Performance Assessment and Checkout Facility (EGNOS)
P-code	Precision code
pdf	portable document format
PDI	Predetection integration interval
PDOP	Position dilution of precision
PI	Proportional and integral (controller)

PID	Process input data (of WAAS); proportional, integral, and differential (control)
PIGA	Pendulous integrating gyroscopic accelerometer
PL	Pseudolite
PLL	Phase-lock loop
PLRS	Position Location and Reporting System (U.S. Army)
PN	Pseudorandom noise
POR	Pacific Ocean Region
PPP	Precise point positioning
PPS	Precise Positioning Service
pps	Pulse per second
PR	Pseudorange
PRC	People's Republic of China
PRN	Pseudorandom noise or pseudorandom number (=SVN for GPS)
PRS	Public Regulated service (of Galileo)
PSD	Power spectral density
Q	Quadrature
QPSK	Quadrature phase shift keying
QZS	Quasi-Zenith Satellite
QZSS	Quasi-Zenith Satellite System
RAAN	right ascension of ascending node
RAG	Receiver antenna gain (relative to isotropic)
RAIM	Receiver autonomous integrity monitoring
RF	Radio frequency
RHCP	Right-hand circularly polarized
RIMS	Ranging and Integrity Monitoring Station(s) (EGNOS)
RINEX	Receiver independent exchange format (for GPS data)
RL	Return loss
RLG	Ring laser gyroscope
RM A	Reliability, maintainability, availability
RMS	Root mean square; reference monitoring station
RNSS	Radio navigation satellite service
RPY	Roll–pitch–yaw (coordinates)
RTCA	Radio Technical Commission for Aeronautics
RTCM	Radio Technical Commission for Maritime Service
RTOS	Real-time operating system
RVCG	Rotational vibratory coriolis gyroscope
s	second
SA	Selective availability (also abbreviated "S/A")
SAIF	Submeter accuracy with integrity function (in QZSS)
SAP	Space adaptive processing
SAR	Synthetic Aperture Radar or Search and Rescue (Galileo service)
SARP	Standards and Recommended Practices (Japan)

SARSAT	Search and rescue satellite—aided tracking
SAW	Surface acoustic wave
SBAS	Space-based augmentation system
SBIRLEO	Space-based infrared low earth orbit
SCOUT	Scripps coordinate update tool
SCP	Satellite Correction Processing (of WAAS)
SDR	Software-defined radio
SF	Scale factor
SFAP	Space-frequency adaptive processing
SFIR	Specific force information receiver
SI	Système International
SIS	Signal in space
SM	Solar magnetic
SNAS	Satellite Navigation Augmentation System (China)
SNR	Signal-to-noise ratio
SOL	Safety of Life Service (of Galileo)
SPNS	Standard precision navigation signal (i.e., for GLONASS)
SPS	Standard Positioning Service
sps	symbols per second
SSBN	Ship Submersible Ballistic Nuclear (US)
STAP	Space–time adaptive processing
STF	Signal Task Force (of Galileo)
STM	State transition matrix
SV	Space vehicle
SVN	Space vehicle number (=PRN number for GPS)
SWR	Standing wave ratio
TCS	Terrestrial communications subsystem (for WAAS)
TCXO	Temperature-compensated Xtat (crystal) oscillator
TDOA	Time difference of arrival
TDOP	Time dilution of precision
TEC	Total electron content
TECU	Total electron content units
TCN	Terrestrial Communication Network
TLM	Telemetry word
TLT	Test Loop Translator
TMBOC	Time-multiplexed BOC
TOA	Time of arrival
TOW	Time of week
TTA	Time to alarm
TTFF	Time to first fix
UDRE	User differential range error
UERE	User-equivalent range error
URE	User range error
USAF	United States Air Force
USN	United States Navy

UTC	Universal Time, Coordinated (or Coordinated Universal Time)
UTM	Universal Transverse Mercator
VAL	Vertical alert limit
VCG	Vibratory Coriolis gyroscope
VDOP	Vertical dilution of precision
VHF	Very high frequency (30–300 MHz)
VOR	VHF omnirange (radionavigation aid)
VRW	Velocity random walk
WAAS	Wide-Area Augmentation System (United States)
WADGPS	Wide-area differential GPS
WGS	World Geodetic System
WMP	WAAS Message Processor
WMS	Wide-area master station
WN	Week number
WNT	WAAS network time
WRE	Wide-area reference equipment
WRS	Wide-area reference station
ZLG	Zero-Lock Gyroscope ("Zero Lock Gyro" and "ZLG" are trademarks of Northrop Grumman Corp.)

1

INTRODUCTION

A book on navigation? Fine reading for a child of six![1]

1.1 NAVIGATION

During the European Age of Discovery, in the fifteenth to seventeenth centuries, the word *navigation* was synthesized from the Latin noun *navis* (ship) and the Latin verb stem *agare* (to do, drive, or lead) to designate the operation of a ship on a voyage from A to B—or the art thereof.

In this context, the word *art* is used in the sense of a *skill*, *craft*, *method*, or *practice*. The Greek word for it is *τεχνυ*, with which the Greek suffix *-λογια* (the study thereof) gives us the word *technology*.

1.1.1 Navigation-Related Technologies

In current engineering usage, the art of getting from A to B is commonly divided into three interrelated technologies:

[1]Truant Officer Agatha Morgan, played by Sara Haden in the 1936 film *Captain January*, starring Shirley Temple and produced by Daryl F. Zanuck for 20th Century Fox Studios.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

- **Navigation** refers to the art of determining the current location of an object—usually a vehicle of some sort, which could be in space, in the air, on land, on or under the surface of a body of water, or underground. It could also be a comet, a projectile, a drill bit, or anything else we would like to locate and track. In modern usage, A and B may refer to the object's current and intended dynamic *state*, which can also include its velocity, attitude, or attitude rate relative to other objects. The practical implementation of navigation generally requires observations, measurements, or sensors to measure relevant variables, and methods of estimating the state of the object from the measured values.
- **Guidance** refers to the art of determining a suitable trajectory for getting the object to a desired *state*, which may include position, velocity, attitude, or attitude rate. What would be considered a "suitable" trajectory may involve such factors as cost, consumables and/or time required, risks involved, or constraints imposed by existing transportation corridors and geopolitical boundaries.
- **Control** refers to the art of determining what actions (e.g., applied forces or torques) may be required for getting the object to follow the desired trajectory.

These distinctions can become blurred—especially in applications when they share hardware and software. This has happened in missile guidance [2], where the focus is on getting to B, which may be implemented without requiring the intermediate locations. The distinctions are clearer in what is called "Global Positioning System (GPS) navigation" for highway vehicles, where

- **Navigation** is implemented by the GPS receiver, which gives the user an estimate of the current location (A) of the vehicle.
- **Guidance** is implemented as *route planning*, which finds a route (trajectory) from A to the intended destination B, using the connecting road system and applying user-specified measures of route suitability (e.g., travel distance or total time).
- **Control** is implemented as a sequence of requested driver actions to follow the planned route.

1.1.2 Navigation Modes

From time immemorial, we have had to solve the problem of getting from A to B, and many solution methods have evolved. Solutions are commonly grouped into five basic navigation modes, listed here in their approximate chronological order of discovery:

- **Pilotage** essentially relies on recognizing your surroundings to know where you are (A) and how you are oriented relative to where you want to be (B). It is older than human kind.

- **Celestial navigation** uses relevant angles between local vertical and celestial objects (e.g., the sun, planets, moons, stars) with known directions to estimate orientation, and possibly location on the surface of the earth. Some birds have been using celestial navigation in some form for millions of years. Because the earth and these celestial objects are moving with respect to one another, accurate celestial navigation requires some method for estimating time. By the early eighteenth century, it was recognized that estimating longitude with comparable accuracy to that of latitude (around half a degree at that time) would require clocks accurate to a few minutes over long sea voyages. The requisite clock technology was not developed until the middle of the eighteenth century, by John Harrison (1693–1776). The development of atomic clocks in the twentieth century would also play a major role in the development of satellite-based navigation.
- **Dead reckoning** relies on knowing where you started from, plus some form of heading information and some estimate of speed and elapsed time to determine the distance traveled. Heading may be determined from celestial observations or by using a magnetic compass. Dead reckoning is generally implemented by plotting lines connecting successive locations on a chart, a practice at least as old as the works of Claudius Ptolemy (~85–168 AD).
- **Radio navigation** relies on radio-frequency sources with known locations, suitable receiver technologies, signal structure at the transmitter, and signal availability at the receiver. Radio navigation technology using land-fixed transmitters has been evolving for nearly a century. Radio navigation technologies using satellites began soon after the first artificial satellite was launched by the former Soviet Union in 1957, but the first global positioning system (GPS) was not declared operational until 1993. Early radio navigation systems relied on electronics technologies, and global navigational satellite system (GNSS) also relies on computer technology and highly accurate clocks. Due to the extremely high speed of electromagnetic propagation and the relative speeds of satellites in orbit, GNSS navigation also requires very precise and accurate timing. It could be considered to be a celestial navigation system using artificial satellites as the celestial objects, with observations using radio navigation aids and high-accuracy clocks.
- **Inertial navigation** is much like an automated form of dead reckoning. It relies on knowing your initial position, velocity, and attitude, and thereafter measuring and integrating your accelerations and attitude rates to maintain an estimate of velocity, position, and attitude. Because it is self-contained and does not rely on external sources, it has the potential for secure and stealthy navigation in military applications. However, the sensor accuracy requirements for these applications can be extremely demanding [26]. Adequate sensor technologies were not developed until

the middle of the twentieth century, and early systems tended to be rather expensive.

These modes of navigation can be used in combination, as well. The subject of this book is a combination of the last two modes of navigation: GNSS as a form of radio navigation, combined with inertial navigation. The key integration technology is Kalman filtering, which also played a major role in the development of both navigation modes.

The pace of technological innovation in navigation has been accelerating for decades. Over the last few decades, navigation accuracies improved dramatically and user costs have fallen by orders of magnitude. As a consequence, the number of marketable applications has been growing phenomenally. From the standpoint of navigation technology, we are living in interesting times.

1.2 GNSS OVERVIEW

There are currently four GNSSs operating or being developed. This section gives an overview; a more detailed discussion is given in Chapter 4.

1.2.1 GPS

The GPS is part of a satellite-based navigation system developed by the U.S. Department of Defense under its NAVSTAR satellite program [9, 11, 12, 14–18, 28–31].

1.2.1.1 GPS Orbits The fully operational GPS includes 31 or more active satellites approximately uniformly dispersed around six circular orbits with four or more satellites each. The orbits are inclined at an angle of 55° relative to the equator and are separated from each other by multiples of 60° right ascension. The orbits are nongeostationary and approximately circular, with radii of 26,560 km and orbital periods of one-half sidereal day (≈11.967 h). Theoretically, three or more GPS satellites will always be visible from most points on the earth's surface, and four or more GPS satellites can be used to determine an observer's position anywhere on the earth's surface 24 h/day.

1.2.1.2 GPS Signals Each GPS satellite carries a cesium and/or rubidium atomic clock to provide timing information for the signals transmitted by the satellites. Internal clock correction is provided for each satellite clock. Each GPS satellite transmits two spread spectrum, L-band carrier signals on two of the legacy L-band frequencies—an L1 signal with carrier frequency $f_1 = 1575.42$ MHz and an L2 signal with carrier frequency $f_2 = 1227.6$ MHz. These two frequencies are integral multiples $f_1 = 1540 f_0$ and $f_2 = 1200 f_0$ of a base frequency $f_0 = 1.023$ MHz. The L1 signal from each satellite uses *binary phase-shift keying* (BPSK), modulated by two *pseudorandom noise* (PRN)

codes in phase quadrature, designated as the C/A-code and P-code. The L2 signal from each satellite is BPSK modulated by only the P(Y)-code. A brief description of the nature of these PRN codes follows, with greater detail given in Chapter 4.

Compensating for Ionosphere Propagation Delays This is one motivation for use of two different carrier signals, L1 and L2. Because delay through the ionosphere varies approximately as the inverse square of signal frequency f (delay $\propto f^{-2}$), the measurable differential delay between the two carrier frequencies can be used to compensate for the delay in each carrier (see Ref. 27 for details).

Code-Division Multiplexing Knowledge of the PRN codes allows users independent access to multiple GPS satellite signals on the same carrier frequency. The signal transmitted by a particular GPS signal can be selected by generating and matching, or correlating, the PRN code for that particular satellite. All PRN codes are known and are generated or stored in GPS satellite signal receivers. A first PRN code for each GPS satellite, sometimes referred to as a *precision code* or *P-code*, is a relatively long, fine-grained code having an associated clock or chip rate of $10f_0 = 10.23$ MHz. A second PRN code for each GPS satellite, sometimes referred to as a *clear* or *coarse acquisition code* or *C/A-code*, is intended to facilitate rapid satellite signal acquisition and handover to the P-code. It is a relatively short, coarser-grained code having an associated clock or chip rate $f_0 = 1.023$ MHz. The C/A-code for any GPS satellite has a length of 1023 chips or time increments before it repeats. The full P-code has a length of 259 days, during which each satellite transmits a unique portion of the full P-code. The portion of P-code used for a given GPS satellite has a length of precisely 1 week (7.000 days) before this code portion repeats. Accepted methods for generating the C/A-code and P-code were established by the satellite developer (Satellite Systems Division of Rockwell International Corporation) in 1991 [10, 19].

Navigation Signal The GPS satellite bit stream includes navigational information on the ephemeris of the transmitting GPS satellite and an almanac for all GPS satellites, with parameters providing approximate corrections for ionospheric signal propagation delays suitable for single-frequency receivers and for an offset time between satellite clock time and true GPS time. The navigational information is transmitted at a rate of 50 baud. Further discussion of the GPS and techniques for obtaining position information from satellite signals can be found in Chapter 4 of Ref. 24.

1.2.1.3 Selective Availability (SA) SA is a combination of methods available to the U.S. Department of Defense to deliberately derate the accuracy of GPS for "nonauthorized" (i.e., non-U.S. military) users during periods of perceived threat. Measures may include pseudorandom time dithering and trun-

cation of the transmitted ephemerides. The initial satellite configuration used SA with pseudorandom dithering of the onboard time reference only [19], but this was discontinued on May 1, 2000.

Precise Positioning Service (PPS) Formal, proprietary service PPS is the full-accuracy, single-receiver GPS positioning service provided to the United States and its allied military organizations and other selected agencies. This service includes access to the encrypted P-code and the removal of any SA effects.

Standard Positioning Service (SPS) without SA SPS provides GPS single-receiver (stand-alone) positioning service to any user on a continuous, world-wide basis. SPS is intended to provide access only to the C/A-code and the L1 carrier.

1.2.1.4 Modernization of GPS GPS IIF, GPS IIR-M, and GPS III are being designed under various contracts (Raytheon, Lockheed Martin). These will have a new L2 civil signal and new L5 signal modulated by a new code structure. These frequencies will improve the ambiguity resolution, ionospheric calculation, and C/A-code positioning accuracy.

1.2.2 Global Orbiting Navigation Satellite System (GLONASS)

A second system for global positioning is the GLONASS, placed in orbit by the former Soviet Union, and now maintained by the Russian Republic [21, 22].

1.2.2.1 GLONASS Orbits GLONASS has 24 satellites, distributed approximately uniformly in three orbital planes (as opposed to six for GPS) of eight satellites each (four for GPS). Each orbital plane has a nominal inclination of 64.8° relative to the equator, and the three orbital planes are separated from each other by multiples of 120° right ascension. GLONASS orbits have smaller radii than GPS orbits, about 25,510 km, and a satellite period of revolution of approximately 8/17 of a sidereal day.

1.2.2.2 GLONASS Signals The GLONASS system uses frequency-division multiplexing of independent satellite signals. Its two carrier signals corresponding to L1 and L2 have frequencies $f_1 = (1.602 + 9k/16)$ GHz and $f_2 = (1.246 + 7k/16)$ GHz, where $k = -7, -6, \ldots 5, 6$ is the satellite number. These frequencies lie in two bands at 1.598–1.605 GHz (L1) and 1.242–1.248 GHz (L2). The L1 code is modulated by a C/A-code (chip rate = 0.511 MHz) and by a P-code (chip rate = 5.11 MHz). The L2 code is presently modulated only by the P-code. The GLONASS satellites also transmit navigational data at a rate of 50 baud. Because the satellite frequencies are distinguishable from

each other, the P-code and the C/A-code are the same for each satellite. The methods for receiving and analyzing GLONASS signals are similar to the methods used for GPS signals. Further details can be found in the patent by Janky [19]. GLONASS does not use any form of SA.

1.2.2.3 Next Generation GLONASS The satellite for the next generation of GLONASS-K was launched on February 26, 2011 and continues to undergo flight tests. This satellite is transmitting a test CDMA signal at a frequency of 1202 MHz.

1.2.3 Galileo

The Galileo system is the third satellite-based navigation system currently under development. Its frequency structure and signal design is being developed by the European Commission's (EC's) Galileo Signal Task Force (STF), which was established by the EC in March 2001. The STF consists of experts nominated by the European Union (EU) member states, official representatives of the national frequency authorities, and experts from the European Space Agency (ESA).

1.2.3.1 Galileo Navigation Services The EU intends the Galileo system to provide the following four navigation services plus one search and rescue (SAR) service.

Open Service (OS) The OS provides signals for positioning and timing, free of direct user charge, and is accessible to any user equipped with a suitable receiver, with no authorization required. In this respect, it is similar to the current GPS L1 C/A-code signal. However, the OS is expected to be of higher quality, consisting of six different navigation signals on three carrier frequencies. OS performance is expected to be at least equal to that of the modernized Block IIR-M GPS satellites, which began launching in 2005, and the future GPS III system architecture currently being developed. OS applications will include the use of a combination of Galileo and GPS signals, thereby improving performance in severe environments such as urban canyons and heavy vegetation.

Safety of Life Service (SOL) The SOL service is intended to increase public safety by providing certified positioning performance, including the use of certified navigation receivers. Typical users of SOL will be airlines and transoceanic maritime companies. The European (also Geostationary) Navigation Overlay System (EGNOS) regional European enhancement of the GPS system will be optimally integrated with the Galileo SOL service to have independent and complementary integrity information (with no common mode of failure) on the GPS and GLONASS constellations. To benefit from

the required level of protection, SOL operates in the L1 and E_5 frequency bands reserved for the Aeronautical Radionavigation Services.

Commercial Service (CS) The CS service is intended for applications requiring performance higher than that offered by the OS. Users of this service pay a fee for the added value. CS is implemented by adding two additional signals to the OS signal suite. The additional signals are protected by commercial encryption and access protection keys are used in the receiver to decrypt the signals. Typical value-added services include service guarantees, precise timing, ionospheric delay models, local differential correction signals for very high-accuracy positioning applications, and other specialized requirements. These services will be developed by service providers, which will buy the right to use the two commercial signals from the Galileo operator.

Public Regulated Service (PRS) The PRS is an access-controlled service for government-authorized applications. It is expected to be used by groups such as police, coast guards, and customs. The signals will be encrypted, and access by region or user group will follow the security policy rules applicable in Europe. The PRS will be operational at all times and in all circumstances, including periods of crisis. A major feature of PRS is the robustness of its signal, which protects it against jamming and spoofing.

SAR The SAR service is Europe's contribution to the international cooperative effort on humanitarian SAR. It will feature near real-time reception of distress messages from anywhere on Earth, precise location of alerts (within a few meters), multiple satellite detection to overcome terrain blockage, and augmentation by the four low earth orbit (LEO) satellites and the three geostationary satellites in the current Cosmitscheskaja Sistema Poiska Awarinitsch-Search and Rescue Satellite (COSPAS-SARSAT) system.

1.2.3.2 Galileo Signal Characteristics Galileo will provide 10 right-hand circularly polarized navigation signals in three frequency bands. The various signals fall into four categories: F/Nav, I/Nav, C/Nav, and G/Nav. The F/Nav and I/Nav signals are used by the OS, CS, and SOL services. The I/Nav signals contain integrity information, while the F/Nav signals do not. The C/Nav signals are used by the CS, and the G/Nav signals are used by the PRS.

E_{5a}–E_{5b} Band This band, which spans the frequency range from 1164 to 1214 MHz, contains two signals, denoted E_{5a} and E_{5b}, which are respectively centered at 1176.45 and 1207.140 MHz. Each signal has an in-phase component and a quadrature component. Both components use spreading codes with a chipping rate of 10.23 Mcps (million chips per second). However, the in-phase components are modulated by navigation data, while the quadrature components, called *pilot signals*, are data-free. The data-free pilot signals permit arbitrarily long coherent processing, thereby greatly improving detection and

tracking sensitivity. A major feature of the E_{5a} and E_{5b} signals is that they can be treated as either separate signals or a single wide-band signal. Low-cost receivers can use either signal, but the E_{5a} signal might be preferred, since it is centered at the same frequency as the modernized GPS L5 signal and would enable the simultaneous reception of E_{5a} and L5 signals by a relatively simple receiver without the need for reception on two separate frequencies. Receivers with sufficient bandwidth to receive the combined E_{5a} and E_{5b} signals would have the advantage of greater ranging accuracy and better multipath performance.

Even though the E_{5a} and E_{5b} signals can be received separately, they actually are two spectral components produced by a single modulation called alternate *binary offset carrier* (altBOC) modulation. This form of modulation retains the simplicity of standard BOC modulation (used in the modernized GPS M-code military signals) and has a constant envelope while permitting receivers to differentiate the two spectral lobes. The current modulation choice is altBOC(15,10), but this may be subject to change.

The in-phase component of the E_{5a} signal is modulated with 50 sps (symbols per second) navigation data without integrity information, and the in-phase component of the E_{5b} signal is modulated with 250 sps data with integrity information. Both the E_{5a} and E_{5b} signals are available to the OS, CS, and SOL services.

E_6 Band This band spans the frequency range from 1260 to 1300 MHz and contains a C/Nav signal and a G/Nav signal, each centered at 1278.75 MHz. The C/Nav signal is used by the CS service and has both an in-phase and a quadrature pilot component using a BPSK spreading code modulation of 5×1.023 Mcps. The in-phase component contains 1000-sps data modulation, and the pilot component is data-free. The G/Nav signal is used by the PRS service and has only an in-phase component modulated by a BOC(10,5) spreading code and data modulation with a symbol rate that is to be determined.

L1–E_1 Band The L1–E_1 band (sometimes denoted as L1 for convenience) spans the frequency range from 1559 to 1591 MHz and contains a G/Nav signal used by the PRS service and an I/Nav signal used by the OS, CS, and SOL services. The G/Nav signal has only an in-phase component with a BOC spreading code and data modulation; the characteristics of both are still being decided. The I/Nav signal has an in-phase and quadrature component. The in-phase component will contain 250-sps data modulation and will use a BOC(1,1) spreading code, but this has not been finalized. The quadrature component is data-free.

1.2.3.3 Updates The first Galileo In-Orbit Validation Element (GIOVE) satellites, designated as GIOVE-A, GIOVE-B, were launched in 1995 and 2008, respectively. The next two were launched in October 2011 and began

broadcasting in December 2011. All Galileo signals were activated on December 17, 2011 simultaneously for the first time across the European GNSS system's three spectral bands known as E_1 (1559–1592 MHz), E_5 (1164–1215 MHz), and E_6 (1260–1300 MHz).

1.2.4 Compass (BeiDou-2)

The BeiDou Navigation Satellite System is being developed by the People's Republic of China (PRC), starting with regional services, and later expanding to global services. Phase I was established in 2000. Phase II is to provide for areas in China and its surrounding areas by 2012. Phase III will provide global service by 2020.

1.2.4.1 Compass Satellites BeiDou will consist of 27 medium earth orbit (MEO) satellites, including 5 geostationary earth orbit (GEO) satellites and 3 inclined GEO satellites. The GEO satellites will be positioned at 58.75°E, 80°E, 110.5°E, 140°E, and 160°E.

1.2.4.2 Frequency The nominal carrier frequency of 1561.098 MHz with B1 signal is currently a quadrature phase-shift key in (QPSK) modulation. Compass now has 13 (as of 8/6/2012) BeiDou-2 satellites operating in its constellation [1]. Details of this section are given in Chapter 4.

1.3 INERTIAL NAVIGATION OVERVIEW

The purpose of this section is to explain how and why inertial navigation came about, how it works in general, and what it is used for—mostly from a historical standpoint. Although the development of inertial systems technology has involved perhaps hundreds of thousands of people, much of their work was so highly classified that little of their history is known today. Historical discussion here is very limited, mostly about those technologies still used today, and with limited references to the multitude of people it took to make it happen. More detail can be found in the historical accounts by Draper [5], Gibson [8], Mackenzie [26], Mueller [28], Wagner [34], and Wrigley [35]. For an account of the contemporary computer hardware and software developments through that period, see McMurran [27].

Chapter 3 has the essential technical details about hardware and software implementations, and Chapter 11 is about analytical methods for statistical characterization of navigation performance.

1.3.1 Theoretical Foundations

It has been called "Newtonian navigation" [33] because its theoretical foundations have been known since the time of Newton:

Given the position $x(t_0)$ and velocity $v(t_0)$ of a vehicle at time t_0, and its acceleration $a(s)$ for times $s > t_0$, then its velocity, $v(t)$, and position, $x(t)$, for all time $t > t_0$ can be defined as

$$v(t) = v(t_0) + \int_{t_0}^{t} a(s)\,ds$$

$$x(t) = x(t_0) + \int_{t_0}^{t} v(s)\,ds.$$

It follows that, given the initial position $x(t_0)$ and velocity $v(t_0)$ of a vehicle, its subsequent position depends only on its subsequent accelerations. If these accelerations could be measured and integrated, this would provide a navigation solution.

However, the technology of Newton's time was inadequate for practical implementation. What was missing included the following:

1. Sensors for measuring acceleration with accuracy sufficient for the intended mission. Two types of sensors would be required:
   (a) acceleration sensors for measuring each of the three components of acceleration
   (b) rotation sensor for keeping track of the directions of the acceleration components being measured from inside a moving vehicle.
2. Compatible methods for integrating the sensor outputs to obtain position and for putting the results into compatible formats for the application. This would include
   (a) integrating the outputs of rotation sensors to determine the orientation of the acceleration sensors
   (b) integrating the measured accelerations to obtain velocities and integrating the velocities to obtain position.
3. Hardware and software for implementing these methods and for putting the results into useable forms.
4. Applications that could justify the investments in technology required for developing the solutions to the capabilities listed above. It could not be justified for transportation at the pace of a sailing ship or a horse, and it would not happen until we had long-range missiles.

These issues are addressed in the remainder of this section.

1.3.2 Inertial Sensor Technology

The following accounts provide a rather small sample of the technologies developed for inertial navigation, covering mostly those that have remained in use today.

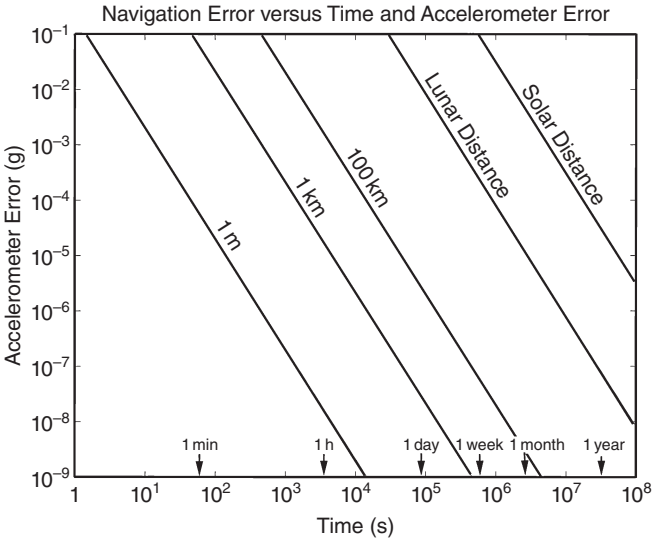


Fig. 1.1 Inertial navigation error as a function of sensor error and time.

1.3.2.1 Sensor Requirements Inertial navigation performance is primarily limited by inertial sensor performance because the sensor accuracies required for achieving even modest navigational performance can be difficult to attain. Figure 1.1 is a contour plot of the evolution over time of inertial navigation position error δ_{pos} resulting from accelerometer (acceleration sensor) error δ_{acc}, using Newton's model:

$$\delta_{\text{pos}} = \frac{1}{2}\delta_{\text{acc}} t^2,$$

where t is the time since navigation began. The plotted results would indicate, for example, that achieving 1 km of navigation accuracy after a week at sea would require acceleration sensor accuracies better than 10^{-9} g ($\approx 9.8 \times 10^{-9}$ m/s/s), and even something as modest as 1 km after an hour would require sensor accuracies in the order of 10^{-5} g. Achieving such accuracies aboard a moving vehicle was not going to be easy.

It turns out that, for terrestrial inertial navigation, a similar plot can be made in which accelerometer error (in units of 1 g) is replaced by attitude sensor error (in radians). An inertial navigation accuracy of 1 km after 1 h would then require attitude sensor accuracies in the order of 10^{-5} rad, or about 2 arc seconds. This is because an attitude error of 10^{-5} rad results in a miscalculation of gravitational acceleration[2] by about 10^{-5} g.

[2]It also turns out that, in the terrestrial environment, the shape of the gravitational field ameliorates the buildup of position errors with time. That subject is discussed in Chapter 11.

1.3.2.2 Motivation Inertial navigation is a product of the Cold War between the Soviet Bloc and NATO Allies. The United States and the Soviet Union had cooperated in defeating Nazi Germany in World War II. At war's end, both sides rushed to capture what they could find in the way of German military technology and those who developed it, and they did not wish to share. It was a harbinger of the coming Cold War.

The United States had developed and used nuclear weapons toward the end of World War II, and the Soviet Union was quick to develop its own nuclear capabilities. Both sides now felt they needed compatible long-range delivery systems, and both sides mounted well-funded programs to develop that capability. During that period, neither side could assume it had the option of doing only what it could afford to do. It was a time for doing whatever one cannot afford not to do. This was what motivated development of inertial navigation.

1.3.2.3 Inertial Sensors Prior to Newton Inertial sensors are actually older than Newton, although little was known about them at the time. Many species of flying insects have an extra pair of modified wings called *halters*, which function as rotation rate sensors during flight. Today, we would recognize the design as a *vibrating Coriolis gyroscope*.

Closer to home, you carry around in your head a pair of even more sophisticated inertial sensor systems. In the bony mass behind each ear is a *vestibular system.* It has been evolving since the time your ancestors were fish [32]. Each of your vestibular systems consists of a set of three rotation sensors (*semicircular canals*), with roughly orthogonal axes of rotational sensitivity. These are augmented by a set of acceleration sensors (*saccule* and *utricle*) indicating the direction and magnitude of accelerations due to forces applied to your head. Your vestibular systems play a major role in compensating your vision system during rotations of your head. They also help you maintain your balance and attitude awareness when you are without visual cues. Their performance is not what we would consider "inertial grade" (i.e., good enough for practical inertial navigation), but they have the essential elements of an inertial navigation system (INS):

1. a complete set of three quasi-orthogonal sensors for changes in the attitude of your head
2. sensors for three orthogonal components of acceleration of your head due to applied forces
3. a (neurological) processor for resolving these sensor outputs into applicable representations of the displacements and attitude changes you have experienced.

Man-made systems for inertial navigation have the same essential parts except that they are not biological (yet).

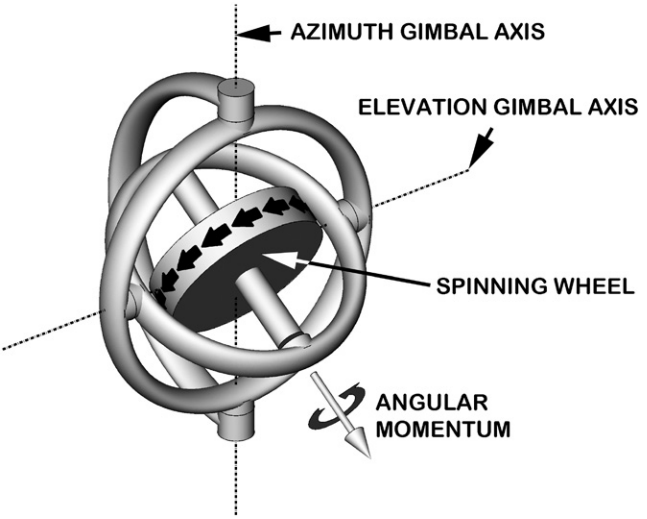


Fig. 1.2 Essential features of the Bohnenberger–Foucault gyroscope.

1.3.2.4 Early Momentum Wheel Gyroscopes (MWGs) The word *gyroscope* was coined by Jean Bernard Léon Foucault (1819–1868) in the mid-nineteenth century. The name was composed from Greek stem-words meaning essentially "rotation sensor," and that is what it still means. In 1852, Foucault used a design like that illustrated in Fig. 1.2 to measure the rotation of the earth [7]. It is called an *MWG* because it uses the conservation of angular momentum of the spinning wheel to maintain an inertially fixed (i.e., nonrotating, ideally) reference direction: the rotor spin axis. Foucault's gimbaled gyroscope was similar to an earlier design by Bohnenberger [3], but with a spinning wheel in place of Bohnenberger's spinning sphere.

Foucault's gyroscope had to be spun up manually and was only useful for a matter of minutes before bearing drag slowed it down. Methods for sustaining rotor spin (using compressed gas or electricity) would solve the run-down problem, and further improvements in bearing technology resulted in significant advances in the ability of MWGs to maintain a true inertial direction. Since Foucault's time, MWG bearings have included

1. thrust or sleeve bearings
2. pivot or jewel bearings
3. ball or roller bearings
4. gas bearings
5. magnetic bearings
6. electrostatic bearings.

Many of these improvements occurred early in the nineteenth century, by which time MWGs would replace the magnetic compass, which was not that reliable aboard iron ships. Soon after aircraft were introduced, MWGs were also being developed for flight instrumentation.

1.3.2.5 German Inertial Technology: 1930s–1945 The Treaty of Versailles restricted development of artillery in Germany. The response was to develop alternative weapons not covered by the Treaty [8]. The Nazi government put great effort into alternative means for delivering explosive projectiles over long distances. From this would come a cruise missile (V-1) and a ballistic missile (V-2). These had greater ranges than artillery, but both required means for autonomous guidance and control during flight. Their developers experimented with magnetic compasses, radio navigation, and inertial sensors.

World War II German technologies for inertial sensing and control were remarkably advanced for the time. Although the related technologies for onboard processing were severely limited, they were adequate for short-range missions of several minutes' duration. Most guidance computations were done on the ground and simplified to a form that could be implemented onboard a rocket or cruise missile using the electromechanical technology of the day. Onboard inertial guidance implementation had to be done "open loop," in the sense that control was used only to follow a preprogrammed trajectory without feedback related to trajectory errors. Control was achieved by using rotation sensors to detect deviations from the intended heading (yaw angle) and pitch angle, and feedback was applied to aerodynamic actuators (ailerons and elevons for cruise missiles, or equivalent vanes in rocket exhaust). Radio navigation was also used as an aid for maintaining the planned heading, and the cruise missile used a barometric altimeter to control altitude. The amount of fuel loaded into the V-1 cruise missile was used as a means for controlling its range. Range control for the V-2 rocket used a rather sophisticated integrating acceleration sensor to keep track of accumulated velocity for initiating thrust termination.

Gyroscopes had been used before this time for the control of torpedoes[3] and rockets.[4] German innovations in inertial technology would include

1. "Inertial platforms," capable of maintaining a nonrotating orientation with respect to the celestial sphere. This would remain an essential component of all INSs for decades. All spacecraft for the Apollo moon missions, for example, contained inertial platforms. Even today, the most accurate INSs use inertial platforms.
2. The first inertial grade integrating accelerometer, the output of which is proportional to accumulated acceleration (velocity change). This basic

[3]In the 1890s, by British torpedo pioneer Robert Whitehead.
[4]In 1932, by American rocket pioneer Robert H. Goddard.

design concept, with implementation refinements, has had different names at different times, but it is still being used today.

3. Electromechanical servomechanisms for feedback control, used in the implementations of the inertial platform, missile guidance, and fire-control systems.
4. Models and methods for predicting inertial system performance based on sensor error characteristics. These models were essential for developing unmanned missiles, with no operator available to observe performance and make corrections. These models enabled systems designers to predict relative performance of alternative designs before systems were flight-tested.

German scientists and engineers were successful in developing inertial guidance and control for missiles, which was all that was needed at the time. Inertial navigation would come later.

The Inertial Platform An inertial platform is a hardware solution to the problem of keeping track of the directions of the acceleration components being measured from inside a moving vehicle. It solves the problem by keeping the acceleration sensors in a known, inertially fixed (i.e., nonrotating) orientation, a concept developed and patented by Johann M. Boykow (1878–1935), who had also worked on aircraft autopilots.

Figure 1.3 is an illustration of the basic design features of an inertial platform, with a nested set of three gimbals used to allow the innermost member to have complete rotational freedom about three axes. Three gyros (represented by blocks with "G" on all faces) on the innermost member are used to sense rotations about their respective input axes (i.e., axes of rotational sensitivity, represented by arrows), and any sensed rotation is nulled using feedback loops with torque motors in the gimbal bearings. Three accelerometers (represented by blocks with "A" on all faces) are used to measure accelerations along their input axes (also represented by arrows).

The inertial sensor design for the V-2 rocket was similar to the configuration shown in the figure, with the top open for easier access during assembly and testing. The sensors looked quite different, but the operational principles were the same.[5]

Integrating Accelerometers The first acceleration sensor that might be considered "inertial grade" was invented in Germany in the late 1930s by Fritz Mueller (1907–2001) [28] and was named (in English) the *Mueller mechanical integrating accelerometer* (MMIA). Improved models have been called *pendulous integrating gyroscopic accelerometers* (PIGAs), or *specific*

[5]MWGs are capable of sensing rotation about two axes, which complicates the implementation just a bit.

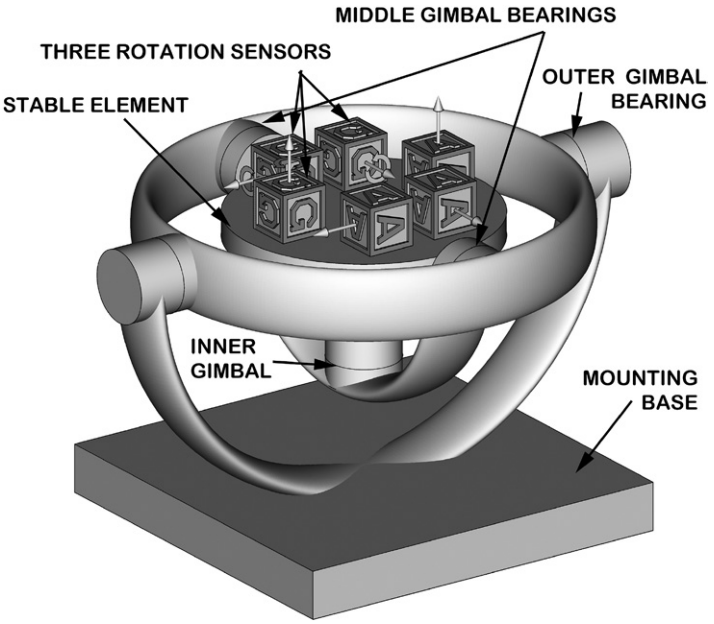


Fig. 1.3 Basic design features of a gimbaled inertial platform.

force integrating receivers (SFIRs, pronounced "siffers"), but the underlying principles have remained the same [13].

The basic features of the MMIA are illustrated in Fig. 1.4. As shown, it is similar in design to the Bohnenberger–Foucault gyroscope illustrated in Fig. 1.2 except that the inner gimbal and rotor assembly has been deliberately unbalanced so that its center of mass is offset from its center of support (along the inner gimbal rotation axis) in the direction along the rotor spin axis. The implementation of the MMIA in hardware requires a beefier gimbal structure than that shown in the figure, and the better implementations use a fluid-filled cylinder as the outer gimbal bearing.

Figure 1.5 illustrates how the MMIA functions. Any force F, applied to the rotor and inner gimbal assembly (including the added mass), is applied through the inner gimbal bearings. Because the inner gimbal rotation axis is offset from the center of mass of the rotor assembly by some distance D, the reactive force ma creates a coupling torque $T = maD$, causing the angular momentum of the rotor assembly to precess about the outer gimbal axis. The angular rate of precession will be proportional to the applied acceleration a, and the total precession angle will be proportional to the integral of acceleration. The MMIA performs this integration rather accurately—more accurately than any of the other methods for implementing integration available in the 1940s. The MMIA used in German V-2 rockets had scale factor errors in the order of

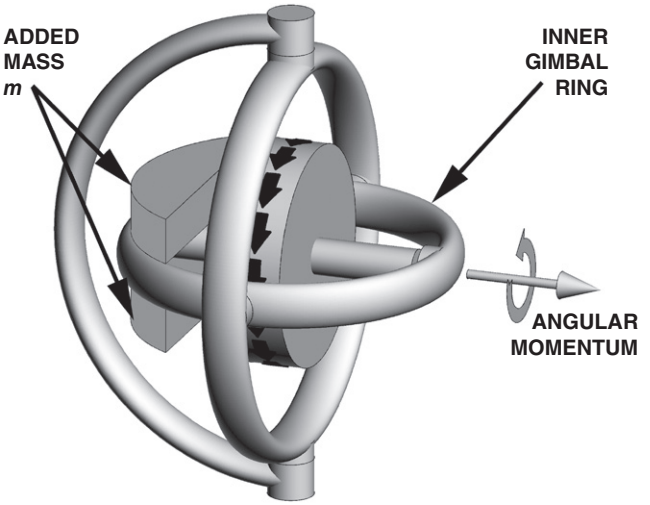


Fig. 1.4 Unbalanced inner gimbal ring of MMIA.

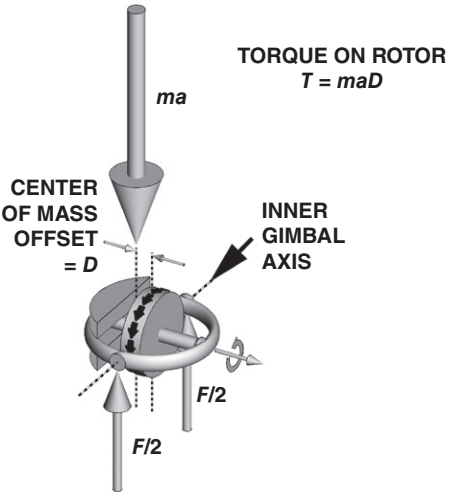


Fig. 1.5 Force–balance mechanics of the MMIA.

0.1%, or a relative error of about 0.001. Today's models have scale factor errors several orders of magnitude smaller.

The MMIA was used to control the range to impact of the V-2 by metering the accumulated acceleration due to thrust and signaling engine cutoff when a preset threshold was reached.

After the war, Mueller came to the United States and continued development of inertial guidance and control systems, including the one that helped put the first American satellite in orbit in 1958.

1.3.2.6 Charles Stark Draper (1901–1987), "The Father of Inertial Navigation" Although, as one of its early pioneers put it, the INS "was apparently evolved rather than invented,"[6] Draper had a lot to do with reducing it to practice. Born before the Wright brothers made their first flight in 1903, he developed an early interest in aviation, obtained a civil aviation license, and flew his own airplane. This experience made it obvious to him that better flight instrumentation was required. His early success in research and development of flight instrumentation led to his founding the Instrumentation Laboratory at MIT in the early 1930s.

During World War II, Draper and his associates at MIT developed antiaircraft fire-control systems used aboard Navy ships and military aircraft. Much of this technology depended on gyroscopic instruments to measure angular rates, analog computation of firing solutions, and servomechanisms to control firing directions.

Beginning in 1946, Draper led a defense-funded research and development project at the Instrumentation Laboratory to develop an inertial navigator for manned missions.[7] Its intended applications included long-range bombers, surface ships, and submarines.

In February of 1953, Draper's team demonstrated an INS aboard a World War II-vintage Boeing B-29 bomber on a flight from Bedford, Massachusetts to Los Angeles, California—a distance of about 2250 nmi (~4167 km). It was the first successful demonstration of acceptable inertial navigation performance over a representative mission distance. Draper's INS was called Space Inertial Reference Equipment (SPIRE). It was the size of a small automobile and weighed around 2700 lb, but it worked.

The Instrumentation Laboratory at MIT would go on to develop successive generations of INSs, including those for the NASA Apollo Command Module and Lunar Excursion Module (LEM), the Air Force Atlas, Thor, Titan, and MX intercontinental ballistic missiles (ICBMs), and the Navy Polaris, Poseidon, and Trident submarine-launched ballistic missiles (SLBM). The Advanced Inertial Reference Sphere (AIRS) designed at MIT for the MX missile is perhaps the most accurate system ever developed for ICBMs. It uses a floated sphere in place of gimbals, called a floated inertial measurement ball (FLIMBAL).

[6]G. R. Pitman, editor, *Inertial Guidance*, Wiley, 1962.

[7]At the time, Russian-born theoretical physicist George Gamow was a member of one of the government science advisory boards and an early critic of Draper's inertial navigation project—on the grounds that inertial navigation was theoretically unstable in vertical navigation. Gamow was correct, but Draper was able to prevail anyway. Navigators aboard surface ships generally knew their altitudes; barometric altimeters would solve the vertical navigation problem for aircraft, and depth sensors would do the same for submarines.

TABLE 1.1. U.S. Ground-Launched Missile Projects Begun in 1946

Project Number	Contractor	Missile Type	Missile Name(s)
MX-770	North American Aviation	Cruise	Navaho
MX-771	Glen L. Martin Co.	Cruise	Matador
MX-772	Curtiss-Wright Corp.	Cruise	[a]
MX-773	Republic Aviation	Cruise	[a]
MX-774	Consolidated Vultee	Ballistic	Hiroc, Atlas
MX-775	Northrup Corp.	Cruise	Snark, Boojum

[a]Cancelled 1947.

The MIT Instrumentation Laboratory was renamed the Charles Stark Draper Laboratory (CSDL) in 1970 and spun off as an independent, MIT-owned, not-for-profit corporation in 1973. It would later play a major role in the development of micron-scale inertial sensors. The designs of all systems developed at MIT and CSDL were turned over to the U. S. Department of Defense, which then put production contracts out for bids by commercial manufacturers.

1.3.2.7 Aerospace Inertial Technology All successful inertial instruments and systems designed at MIT were eventually manufactured by commercial aerospace companies, which helped develop the industry. In addition, however, many aerospace companies were discovering inertia navigation on their own.

Table 1.1 lists some of the major military-funded projects started in 1946 for the purpose of developing practical ground-launched delivery systems for nuclear weapons. All were to be unmanned, requiring automated guidance and control. Most of these were cruise missiles, which were thought to have greater range potential because they could use air-breathing propulsion to eliminate the oxidizer weight needed for rockets. However, as the missile technologies matured, so did nuclear weaponry. By the late 1950s, nuclear payloads had shrunk and rocketry had improved to point that, of these, only the Atlas ballistic missile survived. However, inertial navigation technologies developed for many of the other projects had been so successful that they found applications on other projects.

Project MX-770, for example, was cancelled in 1957, after more than a decade of development. By then, its inertial guidance technology was so advanced it could be successfully applied to inertial navigation of Navy ships, including submarines. A year after Project MX-770 had been terminated, a modified version of the INS developed for Navaho was used for navigating the nuclear submarine *USS Nautilus* under the ice at the North Pole.

By the 1960s, many of the major aerospace and commercial companies were involved in developing inertial sensors and systems. These would come to include such names as Litton, Sperry, Teledyne, Honeywell, and Delco.[8]

[8]Originally named AC Spark Plug, AC being the initials of its founder, Albert Champion (1828–1927).

TABLE 1.2. A Sampling of Inertial Sensor Types

What It Measures	Sensor Type	Physical Phenomenon	Implementation Method
Rotation (gyroscope)	Momentum wheel gyro	Angular momentum	Displacement Torque rebalance
	Coriolis gyro	Coriolis effect	Rotation Vibration
	Optical gyro	Sagnac effect	Fiber-optic gyroscope
		Laser	Ring laser gyro
Acceleration (accelerometer)	Gyroscopic	Precession due to mass unbalance	Displacement Torque rebalance
	Electromagnetic	Induction	Drag cup
		Electromagnetic force	Force rebalance
	Mass–spring	Strain	Piezoelectric Piezoresistance
	Electrostatic	Electrostatic force	Force rebalance

Innovations in inertial technology during this period included a number of new sensor designs based on new physical principles, some of which are listed in Table 1.2.

Improvements in MWGs The major sources of sensor error in MWGs include those due to rotor mass unbalance (the same precession mechanism exploited by the MMIA) and those due to bearing torques.

The mass-unbalance problem was solved by a combination of improved manufacturing tolerances and the ability to calibrate it and compensate for it during operation. It is an acceleration-sensitive effect, and it can be compensated using the accelerations measured by the accelerometers. The most accurate inertial sensors developed during this period would all rely on a combination of high-precision manufacturing, sensor calibration, and run-time compensation.

The effects of bearing torques were reduced significantly by two new bearing technologies. One of these uses gas as a lubricant; another uses electrostatic forces to support the rotor in a vacuum.

Early gas-bearing gyroscopes used compressed gas (much like an air hockey table). Joseph C. Boltinghouse (1909–2009) and John M. Slater (1908–1987) developed an alternative bearing design using precise spherical bearing surfaces in very close proximity, such that no gas pumping was required. The spinning of the rotor was sufficient to maintain gas lubrication of the bearing surfaces. This design was used in the INS for Minuteman missiles.

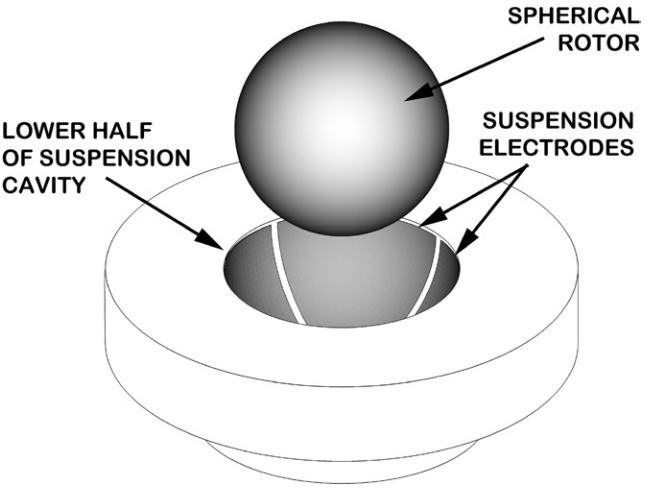


Fig. 1.6 The Boltinghouse micro-ESG.

A further reduction in bearing torques was achieved by using electrostatic pressure[9] to suspend a spherical rotor in a spherical electrode cavity. The electrostatic gyroscope (ESG) was first developed by Arnold T. Nordsieck (1911–1971), then a professor of physics at the University of Illinois, and was first applied to inertial navigation in the 1960s by the Honeywell Corporation. The Nordsieck design uses a hollow beryllium ball about the size of a golf ball, which is somewhat difficult to manufacture to the close tolerances required on its inside and outside surfaces. Autonetics instrument designer Joseph C. Boltinghouse developed an alternative design, illustrated in Fig. 1.6. It has a smaller (1-cm diameter) solid beryllium rotor, which is much easier to manufacture. The U.S. Navy Electrostatically Suspended Gyro Navigator (ESGN), using the Boltinghouse ESG, would be the primary INS aboard Trident-class submarines for decades. Its performance is classified, but it is probably the most accurate INS ever built for submarine navigation.

The most accurate gyroscopes made to date have been ESGs. However, these were not used for inertial navigation, but for a theoretical physics experiment named "Gravity Probe B." This was a NASA-funded program started in 1976 and continued until 2011. Its mission was to resolve two fine points of Einstein's theory of gravitation, the most demanding of which was to measure an effect called "frame dragging," predicted to cause an effective inertial coordinate rotation rate of around 37 milliarcseconds/year (~10^{-9} deg/h). A team at Stanford University led by Dr. Francis Everitt, designed the scientific payload,

[9]It is easier to create negative (attractive) pressure with electrostatics, so the rotor is suspended by overhead electrostatic attraction.

a satellite containing four superconducting ESGs. These were able to measure the frame-dragging rate with a 95% confidence level of about $\pm 2.3 \times 10^{-10}$ deg/h [6].

Optical Gyroscopes The first optical gyroscopes were designed in the 1960s, not long after the first functioning helium neon laser was demonstrated.[10] These were ring laser gyroscopes (RLGs), which use a closed-loop lasing path with mirrors at the corners and laser beams propagating in both directions. The phase coherence of the two counter-rotating beams can be measured by optical interferometry of the two beams, deliberately allowed to leak through one of the corner mirrors. Rotation of the device about an axis orthogonal to the plane of the laser path will (due to the finite velocity of light) cause one beam to advance in phase relative to the other. Early designs were plagued by a phenomenon called "lock-in," in which the two counter-rotating beams would remain phase locked at low rotation rates. RLG designers at Honeywell discovered that lock-in was not instantaneous and were successful in avoiding lock-in by adding zero-mean pseudorandom dither in rotation of the optical element relative to its mounting base. In the 1980s, instrument designers at Litton Guidance & Control Systems (now the Navigation Systems Division of Northrop Grumman) solved the fundamental problem with a design using a combination of dual-frequency lasing and a nonplanar lasing path, as illustrated in Fig. 1.7. Appropriately enough, the design is called the Zero Lock Gyro™, or ZLG™ (both registered trademarks).

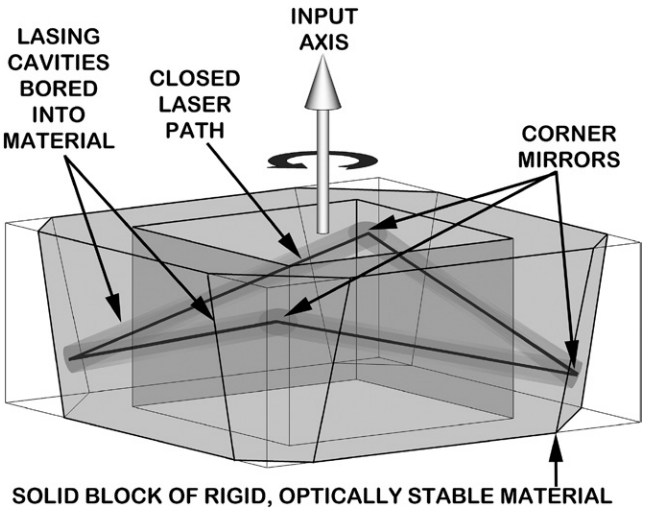


Fig. 1.7 Basic design features of a ring laser gyroscope (RLG).

[10]By A. Javan, W. Bennett, and D. Herriott at Bell Labs in 1960.

RLGs function as *rate-integrating gyroscopes* in that the output phase-shift rate is proportional to the incremental rotation angle. As a consequence, each bit of output represents a fixed rotation angle.

The second fundamental type of optical gyroscope is the fiber-optic gyroscope (FOG). Its development was able to piggyback on commercial optical fiber developments in the 1970s. As illustrated in Fig. 1.8, the FOG uses a common source to transmit laser light both ways through a coil of very long optical fiber. Its function depends on the differential delay due to rotation of the coil and the finite speed of light, a physical phenomenon called the *Sagnac effect*. Optical interferometry of the laser light exiting the fiber at opposite ends will show a phase shift proportional to the rotation rate. As a consequence, the FOG is a rate gyroscope. Its output is proportional to rotation rate.

All optical gyroscope designs require geometric stability to subwavelength levels, and the optical fibers in FOGs are particularly sensitive to stress.

Vibrating Coriolis Gyroscopes The Coriolis effect (derived in Appendix B) is one of the corrections required for modeling Newtonian mechanics in a rotating coordinate frame. It is modeled in the form of an apparent acceleration (*Coriolis acceleration*) experienced by a moving mass whose coordinates are represented in a rotating coordinate system, and has the form

$$a_{\text{Coriolis}} = -2\omega \otimes v_{\text{rotating}},$$

where a_{Coriolis} is the apparent acceleration, ω is the rotation rate (a vector with three components representing the coordinate rotation rates about the three

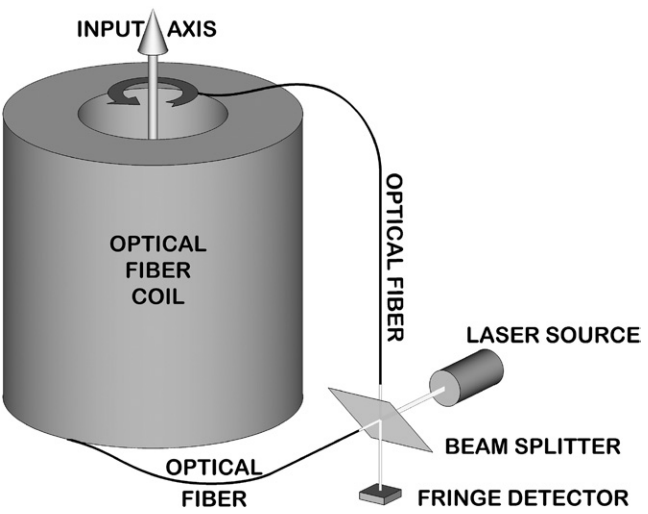


Fig. 1.8 Essential elements of the fiber-optic gyroscope.

rotating coordinate axes), $\otimes$ represents the vector cross product, and v_{rotating} is the velocity of the mass represented in rotating coordinates. The Coriolis effect couples velocity—even vibrational velocity—into acceleration, and the resulting accelerations are orthogonal to the velocities.

The tuning fork gyroscope in Fig. 1.9 illustrates how this effect works to make a rotation rate sensor. The tuning fork is well designed such that the in-and-out motion of its tines is balanced, and no stress is transmitted to the handle. However, when the tuning fork is rotated about its handle, the resulting Coriolis effect couples the balanced in-plane vibration mode into an unbalanced, twisting out-of-plane vibration mode, which produces vibrational torque on the handle. This output vibration can be sensed by using strain sensors between the handle and its holder, but a better solution is to attach another tuning fork handle to handle and end to end such that the twisting vibration mode of the attached tuning fork has the same resonant frequency as that of the in-plane vibration mode of the sensing tuning fork. Perhaps the best models use quartz as the tuning fork material. Quartz is piezoelectric, which means that the vibrational mode of the sensing fork can be controlled electronically. It is also stiff and light, which ups the resonant frequencies (a good thing), and practically lossless, which gives the resonator a high Q-factor (another good thing).

Balancing the vibration modes of a Coriolis gyroscope is very important, because any vibration transmitted through its support translates into energy loss (a bad thing) and potential signal coupling with other vibration sensors. Better balance overall can be achieved by using the three-dimensional rotational equivalent of a tuning fork: the **wine glass**. The vibrational modes of wine glasses have been known for some time. In 1890, George H. Bryan (1864–1928) discovered that, when a vibrating wine glass is rotated about its

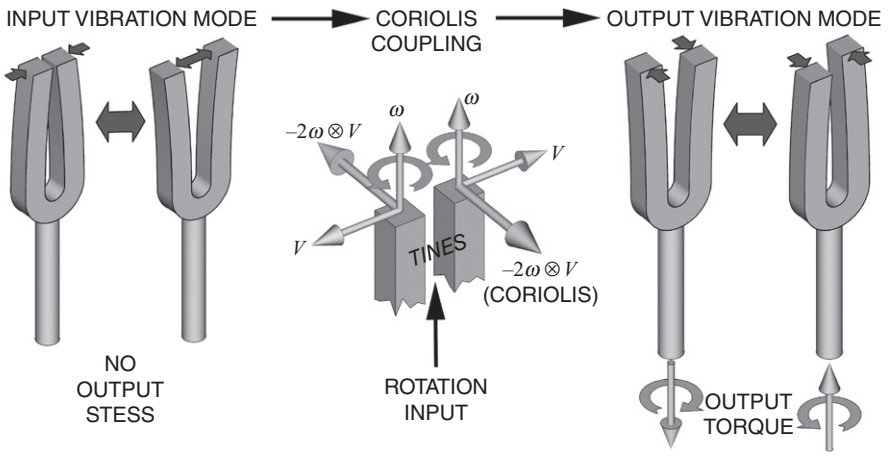


Fig. 1.9 Vibration modes of the tuning fork gyroscope.

stem, the vibrational nodes on its rim rotate at a rate different from that of the input rotation. Bryan called it the "wave inertia effect." It is now called the **Bryan effect**. Early gyroscopes based on the Bryan effect were called "wine glass gyros" (not a particularly prestigious name). They are now called "**hemispherical resonator gyroscopes**" (HRGs). The Cassini spacecraft sent to Saturn in 1997 uses hemispherical resonator gyroscopes for controlling its orientation.

Microelectromechanical systems (MEMS) are tiny electromechanical devices fabricated using wafer-scale processes based on semiconductor manufacturing technology. MEMS technologies were first developed in the 1970s, and an early application was for the acceleration sensors used in detecting automobile collisions for initiating air-bag deployment. Vibration frequencies of structures tend to increase significantly as the structural dimensions decrease, which meant that MEMS devices could be made to vibrate at frequencies up to ~10^5 Hz. Vibrational velocity scales up with frequency, which meant that the Coriolis effect was strong in MEMS devices.

In the early 1990s the CSDL developed a MEMS gyroscope resembling a slice through a tuning fork, as illustrated in Fig. 1.10. It has a pair of thin rectangular masses like cross sections of the tines, and they are driven to vibrate 180° out of phase, just like the tines of the tuning fork. The difference is that the input rotation axis is in the plane of the structure, and the output vibration mode is normal to the plane of the substrate, so that one mass moves upward while the other moves downward. This mode is detected by sensing capacitive changes between the vibrating masses and the underlying surface. In-plane vibration is controlled using opposing pairs of "comb drives," electrostatic force transducers developed at the University of California at Berkeley. The supporting electronics fit into a single application-specific integrated circuit (ASIC) chip. The Draper tuning fork gyro was licensed and further developed

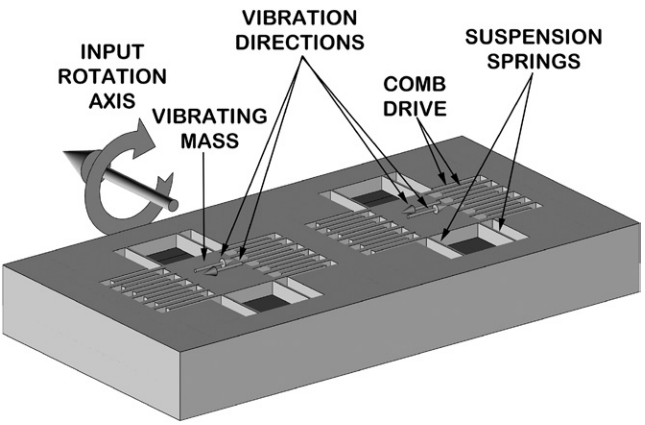


Fig. 1.10 Essential features of the CSDL MEMS gyro.

by Honeywell, which also produces vibrating Coriolis gyroscopes using the plate thickness vibrating mode at ~10^5 Hz.

Another gyroscope design resembles a slice through a HRG, just as the Draper gyro resembles a slice through a tuning fork. The Coriolis effect causes the nodes of the vibrating modes of the resulting ring structure to precess when the device is rotated about the axis normal to the plane of the ring.

Proof Mass Accelerometers A proof mass accelerometer measures the force F required to keep an otherwise isolated "proof mass" m from moving relative to its enclosure. If its enclosure is being accelerated at a rate a due to forces applied to it, then, knowing the force F it is applying to its proof mass m, a can be calculated as

$$a = \frac{F}{m},$$

the force per unit mass, also called *specific force*.

Proof mass accelerometers were developed as alternatives to the MMIA integrating gyroscopic accelerometer mentioned above. The gyroscope in the MMIA makes it sensitive to rotation, which is why it had to be mounted on an inertially stabilized base. It is also rather expensive.

The electromagnetic accelerometer (EMA) is a popular proof mass force–rebalance accelerometer using what is essentially a permanent-magnet speaker drive, as illustrated in Fig. 1.11. The cylindrical speaker coil is mounted on what

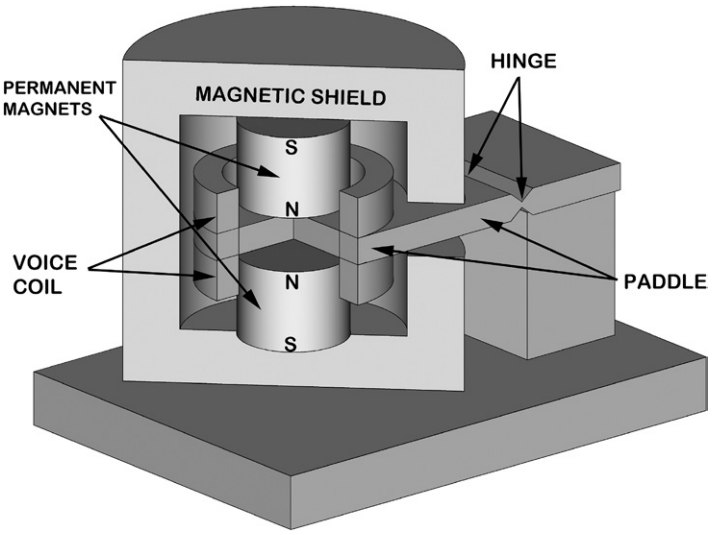


Fig. 1.11 Essential design elements of electromagnetic accelerometer.

is called a "paddle" attached to a compliant hinge, and the current through the coil is servoed to prevent the paddle from moving relative to its enclosure. The measured current is then proportional to the specific force being applied to the proof mass.

MEMS proof mass accelerometers are mostly mass–spring accelerometers, measuring the stress in the structure supporting a proof mass as a measure of the applied acceleration. Many use piezoresistive or piezoelectric films for measuring surface strain. Capacitance variation can also be used to measure displacement, although capacitors also create electrostatic forces corrupting the force measurement.

INS Signal Processing Hardware There was really no computer industry until the early 1950s and no flightworthy computers for inertial navigation until the 1960s. Inertial system implementations in the 1950s used a variety of interim technologies, including digital differential analyzers (DDAs) and magnetic drum memories. DDAs are integrated circuits specifically designed for integration. Enormous effort had to be put into making do with the processor technology of the time. One hybrid missile computer of that era used 128 DDAs together with a small general-purpose computer with only 3 kB of active memory [29].

Silicon transistors and integrated circuits began to revolutionize computer technology in the 1960s. The Apollo moon missions (1969–1972) used onboard computers with magnetic core memories. Magnetic core memory would dominate the market until semiconductor memories appeared in the 1970s. By that time, the cost of magnetic core memory had gotten down to pennies per bit. Memory prices would fall by several orders of magnitude over the next few decades. The introduction of the microprocessor in 1971 marked the beginning of a major downslide in the cost of computing.

Strapdown Systems Strapdown[11] systems use software to replace gimbals by processing the gyro outputs to maintain the coordinate transformation between accelerometer-fixed coordinates and inertial coordinates. The accelerometer outputs can then be transformed to inertial coordinates and processed just as they had been with a gimbaled system—without requiring an inertial platform.

Faster, cheaper computers enabled the development of strapdown inertial technology. Some vehicles (e.g., torpedoes) had been using strapdown gyroscopes for steering control since the late nineteenth century, but now they could be integrated with accelerometers to make a strapdown INS. This eliminated the expense of gimbals, but it also required considerable progress in attitude estimation algorithms [4]. Computers also enabled "modern"

[11]The terminology refers to the idea that the inertial sensors can be "strapped down" to the vehicle frame, although some form of vibration isolation is generally required.

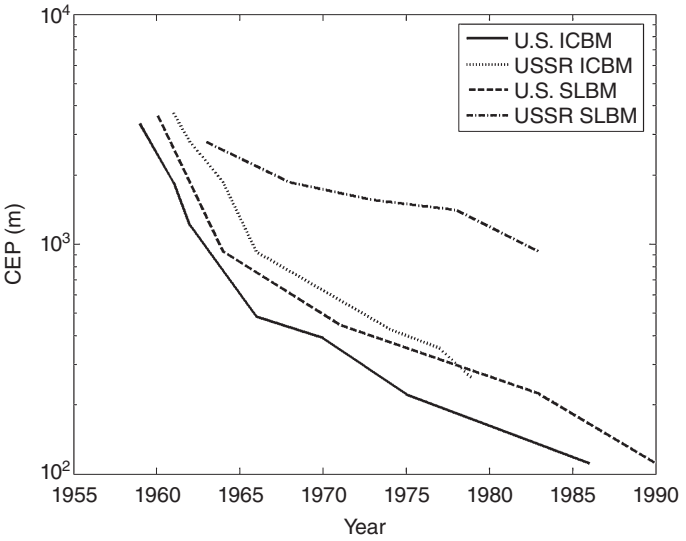


Fig. 1.12 Cold War missile accuracy improvements.

estimation and control, based on state space models. This would have a profound effect on sensor integration capabilities for INS.

A gimbaled INS was carried on each of nine Apollo command modules from the earth to the moon and back between December 1968 and December 1972, but a strapdown INS was carried on each of the six[12] Lunar Excursion Modules (LEMs) that shuttled two astronauts from lunar orbit to the lunar surface and back.

By the mid-1970s, strapdown systems were able to demonstrate navigational accuracies in the order of 1 nmi/h (CEP[13] rate). This was considered adequate for commercial aircraft at that time.

The Race for Accuracy Figure 1.12 is a plot of strategic missile capabilities developed during the Cold War, in terms of inertial guidance accuracies achieved over that period. Expected miss distances are given in meters, CEP. The different plots are labeled according to whether they are for U.S. or USSR missiles, and whether they are for surfaced-launched ICBMs or SLBMs. These data are from Tables A.1 and A.2 of Ref. 26, which should be consulted for

[12]Two additional LEMs were carried to the moon but did not land there. The Apollo 13 LEM did not make its intended lunar landing but played a far more vital role in crew survival.
[13]CEP is an acronym for "circle of equal probability." It is the radius of the 50% confidence circle.

additional clarifying information. According to Mackenzie [26], contributions of the inertial systems to these miss distances are actually rather minor.

1.3.2.8 Developments Since the Cold War The Cold War ended around the time GPS was becoming operational. By that time, INS technology had matured considerably. Not only had achievable accuracies improved by orders of magnitude but so had cost, weight, and power requirements. As a consequence, markets had expanded beyond strategic military applications to include tactical and commercial applications.

The availability of GNSS allowed for integrated GNSS/INS navigation systems accurate enough for automated grading, plowing, and mining. It has also lowered costs to the point that GNSS/INS systems can be embedded in consumer products.

1.4 GNSS/INS INTEGRATION OVERVIEW

1.4.1 The Role of Kalman Filtering

It has been called "navigation's integration workhorse" [23] for the essential role it has played in navigation and especially for integrating different navigation modes. Ever since its introduction in 1960 [20], the Kalman filter has played a major role in the design and implementation of most new navigation systems as a statistically optimal method for estimating position using noisy measurements. Because the filter also produces an estimate of its own accuracy, it has also become an essential part of a methodology for the optimal design of navigation systems. The Kalman filter has been essential for the design and implementation of every GNSS. It is unlikely that the first GNSS (GPS) could have been built without it.

Using the Kalman filter, navigation systems designers have been able to exploit a powerful synergism between GNSSs and INSs, which is possible because they have very complementary error characteristics:

- Short-term position errors from the INS are relatively small, but they degrade significantly over time.
- GNSS position accuracies, on the other hand, are not as good over the short term, but they do not degrade with time.

The Kalman filter takes advantage of these characteristics to provide a common, integrated navigation implementation with performance superior to that of either subsystem (GNSS or INS). By using statistical information about the errors in both systems, it is able to combine a system with tens of meters position uncertainty (GNSS) with another system whose position uncertainty degrades at kilometers per hour (INS) and achieve bounded position uncertainties in the order of centimeters (with differential GNSS) to meters.

1.4.2 Implementation

The Kalman filter solves for the solution with the least mean-squared error by using data weighting proportional to statistical information content (the inverse of uncertainty) in the measured data. It combines GNSS and INS information to

1. track drifting parameters of the sensors in the INS, so that INS performance does not degrade with time when GNSS is available
2. improve overall performance even when there are insufficient satellite signals for obtaining a complete GNSS solution
3. allow the INS to navigate with improved initial error whenever GNSS signals become unavailable
4. improve GNSS signal reacquisition when GNSS signals become available again by providing better navigation solutions (based on INS data)
5. use acceleration and attitude rate information from the INS for reducing the signal phase-tracking filter lags in the GNSS receiver, which can significantly improve GNSS reliability during periods of high maneuvering, jamming, or reduced signal availability.

The more intimate levels of GNSS/INS integration necessarily penetrate deeply into each of the subsystems in that it makes use of partial results that are not ordinarily accessible to users. To take full advantage of the offered integration potential, we must delve into technical details of the designs of both types of systems.

1.4.3 Applications

1.4.3.1 Military Applications The rationale for developing the Navistar GPS system was based, in part, on economic considerations—in terms of how many inertial systems it could replace. However, the ability to integrate GPS with INS also enabled military applications that were not possible before. It would lead to a new generation of high-precision military weaponry, improving military effectiveness while reducing collateral damage. Most missiles were already using inertial sensors for guidance and control, so the transition to integrated GNSS/INS navigation was natural.

Most military applications of inertial navigation were already using other navigation aids for limiting the growth of inertial navigation errors with time. The U.S. Navy had begun using satellites for aiding shipboard inertial navigation decades before GNSS became available, and most military INSs were being adapted to use GPS before it was operational. It has resulted in superior navigation performance at low marginal cost.

1.4.3.2 Civilian and Commercial Applications The availability of GNSS also allowed for integrated GNSS/INS navigation systems accurate enough

for automated grading, plowing, and surface mining. The resulting relaxation of inertial sensor stability requirements and advances in fabrication technologies have also combined to lower costs to the point where low-performance GNSS/INS systems can be embedded in high-end consumer products. This market is likely to grow even more as costs fall due to increasing production volumes.

Details of Section 1.4 are given in Chapters 10 through 12.

PROBLEMS

1.1 How many satellites and orbit planes exist for GPS, GLONASS, and Galileo? What are the respective orbit plane inclinations?

1.2 List the differences in signal characteristics between GPS, GLONASS, and Galileo.

1.3 What are the reference points for GNSS and INS navigators? That is, when one of these produces a position estimate, what part of the respective system is that the position of?

REFERENCES

[1] "BeiDou Navigation Satellite System Signal in Space, Interface control Document," China Satellite Navigation Office, December 2011.

[2] D. J. Biezad, *Integrated Navigation and Guidance Systems*. American Institute of Aeronautics and Astronautics, New York, 1999.

[3] J. G. F. Bohnenberger, "Beschreibung einer Maschine zur Erläuterung der Geseze der Undrehung der Erde um ihre Axe, und der Verländerung der Lage der Letzteren," *Tübinger Blätter für Naturwissenschaften und Arzneikunde* Tübingen, Germany, **3**, 72–83 (1817)

[4] J. E. Bortz, "A New Mathematical Formulation for Strapdown Inertial Navigation," *IEEE Transactions on Aerospace and Electronic Systems* **AES-6**, 61–66 (1971).

[5] C. S. Draper, "Origins of Inertial Navigation," *AIAA Journal of Guidance and Control* **4**(5), 449–456 (1981).

[6] C. W. F. Everitt, D. B. DeBra, B. W. Parkinson, J. Turneaure, J. W. Conklin, M. I. Heifetz, G. M. Keiser, A. S. Silbergleit, T. Holmes, J. Kolodziejczak, M. Al-Meshari, J. C. Mester, B. Muhlfelder, V. G. Solomonik, K. Stahl, P. W. Worden, Jr., W. Bencze, S. Buchman, B. Clarke, A. Al-Jadaan, H. Al-Jibreen, J. Li, J. A. Lipa, J. M. Lockhart, B. Al-Suwaidan, M. Taber, and S. Wang, "Gravity Probe B: Final Results of a Space Experiment to Test General Relativity," *Physical Review Letters* **106**, pp. 221101-1–5 (2011).

[7] L. Foucault, "Sur les phénomènes d'orientation des corps tournants entraînés par un axe fixe à la surfaces de la terre," *Comptes Rendus Hebdomadaires des Seances de l'Academie des Sciences* **35**, 424–427 (1852).

[8] J. N. Gibson, *The Navaho Missile Project: The Story of the "Know-How" Missile of American Rocketry*. Schiffer Military/Aviation History, Atglen, PA, 1996.

[9] *Global Positioning System, Selected Papers on Satellite Based Augmentation Systems (SBASs) ("Redbook")*, Vol. VI. ION, Alexandria, VA, 1999.

[10] "GPS Interface Control Document ICD-GPS-200," Rockwell International Corporation, Satellite Systems Division, Revision B, July 3, 1991.

[11] T. A. Herring, "The Global Positioning System," *Scientific American*, February 1996, pp. 44–50.

[12] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *GPS: Theory and Practice*. Springer-Verlag, Vienna, 1997.

[13] R. E. Hopkins, F. K. Mueller, and W. Haeussermann, "The Pendulous Integrating Gyroscope Accelerometer (PIGA) from the V-2 to Trident D5, the Strategic Instrument of Choice," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Montreal, Canada, August 6–9, 2001.

[14] Institute of Navigation, *Monographs of the Global Positioning System: Papers Published in Navigation ("Redbook")*, Vol. I. ION, Alexandria, VA, 1980.

[15] Institute of Navigation, *Monographs of the Global Positioning System: Papers Published in Navigation ("Redbook")*, Vol. II. ION, Alexandria, VA, 1984.

[16] Institute of Navigation, *Monographs of the Global Positioning System: Papers Published in Navigation ("Redbook")*, with Overview by R. Kalafus, Vol. III. ION, Alexandria, VA, 1986.

[17] Institute of Navigation, *Monographs of the Global Positioning System: Papers Published in Navigation ("Redbook")*, with Overview by R. Hatch, Vol. IV. ION, Alexandria, VA, 1993.

[18] Institute of Navigation, *Monographs of the Global Positioning System: Papers Published in Navigation ("Redbook")*, Vol. V. ION, Alexandria, VA, 1998.

[19] J. M. Janky, *Clandestine Location Reporting by a Missing Vehicle*, U.S. Patent 5629693, May 13, 1997.

[20] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *ASME Transactions, Series D: Journal of Basic Engineering* **82**, 35–45 (1960).

[21] M. Kayton and W. L. Fried, *Avionics Navigation Systems*, 2nd ed. Wiley, New York, 1997.

[22] A. Leick, "*GPS: Satellite Surveying*, 2nd ed. Wiley, New York, 1995, pp. 534–537.

[23] J. J. Levy, "The Kalman Filter: Navigation's Integration Workhorse," *GPS World*, September 1997, pp. 65–71.

[24] T. Logsdon, *The NAVSTAR Global Positioning System*. Van Nostrand Reinhold, New York, 1992.

[25] P. F. MacDoran (inventor), *Method and Apparatus for Calibrating the Ionosphere and Application to Surveillance of Geophysical Events*, U.S. Patent 4463357, July 31, 1984.

[26] D. Mackenzie, *Inventing Accuracy: A Historical Sociology of Nuclear Missile Guidance*. MIT Press, Cambridge, MA, 2001.

[27] M. W. McMurran, *Achieving Accuracy: A Legacy of Computers and Missiles*. Xlibris, Bloomington, IN, 2008.

[28] F. K. Mueller, "A History of Inertial Navigation," *Journal of the British Interplanetary Society* **38**, 180–192 (1985).

[29] B. W. Parkinson and J. J. Spilker, Jr. (Eds.), *Global Positioning System: Theory and Applications*, Vol. 1, Progress in Astronautics and Aeronautics (series). American Institute of Aeronautics and Astronautics, Washington, DC, 1996.

[30] B. W. Parkinson and J. J. Spilker, Jr. (Eds.), *Global Positioning System: Theory and Applications*, Vol. 2, Progress in Astronautics and Aeronautics (series). American Institute of Aeronautics and Astronautics, Washington, DC, 1996.

[31] B. W. Parkinson, M. L. O'Connor, and K. T. Fitzgibbon, "Aircraft Automatic Approach and Landing Using GPS," in B. W. Parkinson and J. J. Spilker, Jr. (Eds.), Chapter 14 in *Global Positioning System: Theory & Applications*, Vol. II, Progress in Astronautics and Aeronautics (series), Vol. 164, Paul Zarchan editor-in-chief. American Institute of Aeronautics and Astronautics, Washington, DC, 1995, pp. 397–425.

[32] N. Shubin, *Your Inner Fish: A Journey into the 3.5-Billion-Year History of the Human Body*. Random House, NY, 2009.

[33] J. M. Slater, *Newtonian Navigation*, 2nd ed. Autonetics Division of Rockwell International, Anaheim, CA, 1967.

[34] J. F. Wagner, "From Bohnenberger's Machine to Integrated Navigation Systems, 200 Years of Inertial Navigation," in Dieter Fritsch (Ed.), *Photogrammetric Week 05*. Wichmann Verlag, Heidelberg, 2005.

[35] W. Wrigley, "History of Inertial Navigation," *Navigation: Journal of the Institute of Navigation* **24**, 1–6 (1977). Baltimore.

2

FUNDAMENTALS OF SATELLITE NAVIGATION SYSTEMS

2.1 NAVIGATION SYSTEMS CONSIDERED

This book is about global navigation satellite systems (GNSS) and inertial navigation systems (INS) and their integration. An INS can be used anywhere on the globe, but it must be updated to remain accurate by independent navigation sources such as GNSS or celestial navigation. Thousands of self-contained INS units are in continuous use on military vehicles, and an increasing number are being used in civilian applications.

2.1.1 Systems Other than GNSS

GNSS signals may be replaced by long-range navigation (LORAN) signals produced by three or more LORAN signal sources positioned at fixed, known locations for location determination (LD). A LORAN system relies on a plurality of ground-based signal towers, preferably spaced several hundreds of kilometers apart, that transmits distinguishable electromagnetic signals that are received and processed by a LORAN signal antenna and a receiver/processor that are analogous to the satellite positioning system signal antenna and receiver/processor. A representative LORAN-C system is discussed in the U.S. Department of Transportation *LORAN-C User's Handbook* [1]. LORAN-C signals use carrier frequencies of the order of 100 kHz and have maximum

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

reception distances of hundreds of kilometers. The combined use of cellular signals for LD inside a building or similar structure can also provide a satisfactory LD system in most urban and suburban communities.

There are other ground-based radiowave signal systems suitable for use as part of an LD system. These include Tacan, U.S. Air Force Joint Tactical Information Distribution System Relative Navigation (JTIDS Relnav), and U.S. Army Position Location and Reporting System (PLRS) (see summaries in Ref. 2, pp. 6–7 and 35–60).

2.1.2 Comparison Criteria

The following criteria may be used in selecting navigation systems appropriate for a given application system:

1. navigation method(s) used
2. system reliability/integrity
3. navigational accuracy
4. region(s) of coverage/availability
5. required transmission frequencies and bands of operation
6. navigation fix update rate
7. user set cost
8. status of system development and readiness.

2.2 SATELLITE NAVIGATION

The Global Positioning System (GPS) is widely used in navigation. Its augmentation with other space-based satellites is the future of near-earth navigation.

2.2.1 Satellite Orbits

GPS satellites occupy six orbital planes inclined 55° from the equatorial plane, as illustrated in Figs. 2.1 and 2.2. Each of the six orbit planes in Fig. 2.2 contains four or more satellites.

2.2.2 Navigation Solution (Two-Dimensional Example)

To start the 3D capability of GNSS, consider the determination of an antenna location in 2D using range measurements [3].

2.2.2.1 Symmetric Solution Using Two Transmitters on Land In this case, the receiver and two transmitters are located in the same plane, as shown in

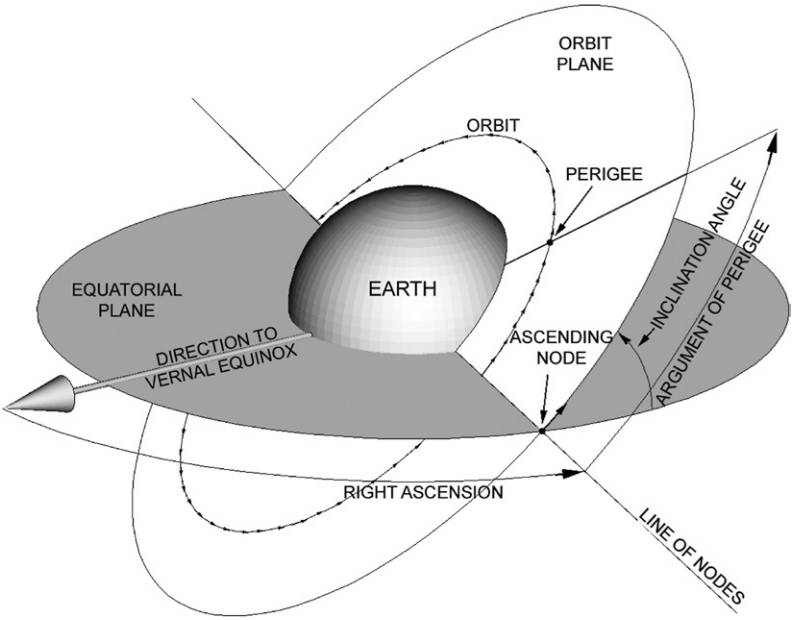


Fig. 2.1 Parameters defining satellite orbit geometry.

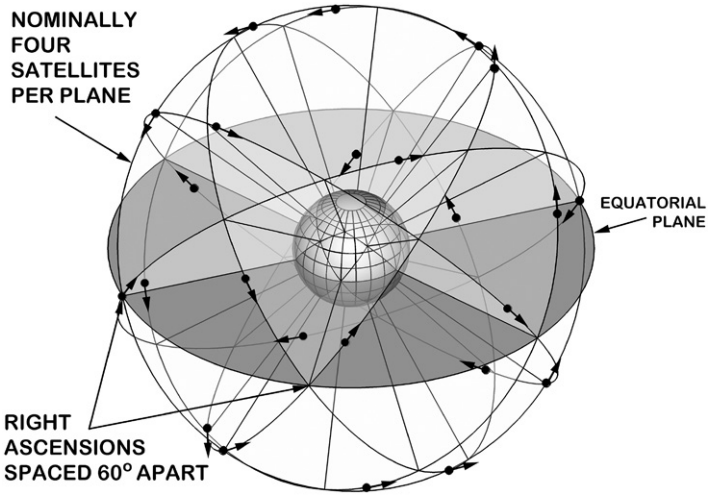


Fig. 2.2 Six GPS orbit planes inclined 55° from the equatorial plane.

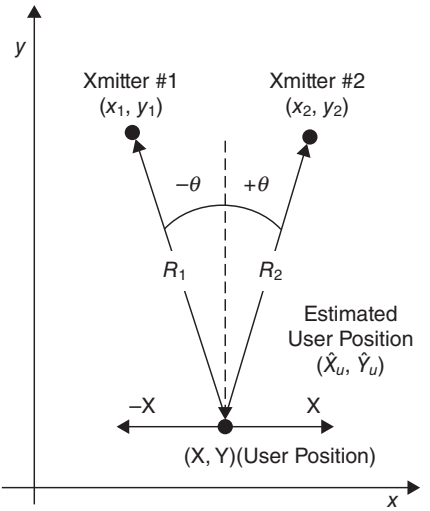


Fig. 2.3 Two transmitters with known 2D positions.

Fig. 2.3, with known transmitter positions x_1, y_1, and x_2, y_2. Ranges R_1 and R_2 of two transmitters from the user position are calculated as

$$R_1 = c\Delta T_1, \tag{2.1}$$

$$R_2 = c\Delta T_2, \tag{2.2}$$

where

c = speed of light (0.299792458 m/ns),
ΔT_1 = time taken for the radiowave to travel from transmitter 1 to the user,
ΔT_2 = time taken for the radiowave to travel from transmitter 2 to the user,
X, Y = user position.

The range to each transmitter can be written as

$$R_1 = \left[(X - x_1)^2 + (Y - y_1)^2\right]^{1/2}, \tag{2.3}$$

$$R_2 = \left[(X - x_2)^2 + (Y - y_2)^2\right]^{1/2}. \tag{2.4}$$

Expanding R_1 and R_2 in Taylor series expansion with small perturbation in X by Δx and Y by Δy yields

$$\Delta R_1 = \frac{\partial R_1}{\partial X}\Delta x + \frac{\partial R_1}{\partial Y}\Delta y + u_1, \tag{2.5}$$

$$\Delta R_2 = \frac{\partial R_2}{\partial X}\Delta x + \frac{\partial R_2}{\partial Y}\Delta y + u_2, \tag{2.6}$$

where u_1 and u_2 are higher-order terms. The derivatives of Eqs. 2.3 and 2.4 with respect to X, Y are substituted into Eqs. 2.5 and 2.6, respectively.

Thus, for the symmetric case, we obtain

$$\Delta R_1 = \frac{X - x_1}{\left[(X - x_1)^2 + (Y - y_1)^2\right]^{\frac{1}{2}}}\Delta x + \frac{Y - y_1}{\left[(X - x_1)^2 + (Y - y_1)^2\right]^{\frac{1}{2}}}\Delta y + u_1, \tag{2.7}$$

$$= \sin\theta\Delta x + \cos\theta\Delta y + u_1, \tag{2.8}$$

$$\text{and} \quad \Delta R_2 = -\sin\theta\Delta x + \cos\theta\Delta y + u_2. \tag{2.9}$$

To obtain the least-squares estimate of (X, Y), we need to minimize the quantity

$$J = u_1^2 + u_2^2, \tag{2.10}$$

which is

$$J = \left(\underbrace{\Delta R_1 - \sin\theta\Delta x - \cos\theta\Delta y}_{u_1}\right)^2 + \left(\underbrace{\Delta R_2 + \sin\theta\Delta x - \cos\theta\Delta y}_{u_2}\right)^2. \tag{2.11}$$

The solution for the minimum can be found by setting $\partial J/\partial\Delta x = 0 = \partial J/\partial\Delta y$, then solving for Δx and Δy:

$$0 = \frac{\partial J}{\partial\Delta x} \tag{2.12}$$

$$= 2(\Delta\mathrm{R}_1 - \sin\theta\Delta\mathrm{x} - \cos\theta\Delta\mathrm{y})(-\sin\theta) + 2(\Delta\mathrm{R}_2 + \sin\theta\Delta\mathrm{x} - \cos\theta\Delta\mathrm{y})(\sin\theta) \tag{2.13}$$

$$= \Delta R_2 - \Delta R_1 + 2\sin\theta\Delta x, \tag{2.14}$$

with solution

$$\Delta x = \frac{\Delta R_1 - \Delta R_2}{2\sin\theta}. \tag{2.15}$$

The solution for Δy may be found in similar fashion as

$$\Delta y = \frac{\Delta R_1 + \Delta R_2}{2\cos\theta}. \tag{2.16}$$

2.2.2.2 Navigation Solution Procedure Transmitter positions x_1, y_1, x_2, y_2 are given. Signal travel times ΔT_1 ΔT_2 are given. Estimated user positions $\hat{X}_u \hat{Y}_u$ are assumed. Set position coordinates X, Y equal to their initial estimates:

$$X = \hat{X}_u, \quad Y = \hat{Y}_u.$$

Compute the range errors:

$$\Delta R_1 = \overbrace{\left[\left(\hat{X}_u - x_1\right)^2 + \left(\hat{Y}_u - y_1\right)^2\right]^{1/2}}^{\text{Estimated ranges}} - \overbrace{C\Delta T_1}^{\text{Measured pseudoranges}}, \tag{2.17}$$

$$\Delta R_2 = \overbrace{\left[\left(\hat{X}_u - x_2\right)^2 + \left(\hat{Y}_u - y_2\right)^2\right]^{1/2}} - C\Delta T_2. \tag{2.18}$$

Compute the theta angle (see Fig. 2.3):

$$\theta = \tan^{-1} \frac{\hat{X}_u - x_1}{\hat{Y}_u - y_1} \tag{2.19}$$

$$= \sin^{-1} \frac{\hat{X}_u - x_1}{\sqrt{\left(\hat{X}_u - x_1\right)^2 + \left(\hat{Y}_u - y_1\right)^2}}. \tag{2.20}$$

Compute user position corrections:

$$\Delta x = \frac{1}{2\sin\theta}(\Delta R_1 - \Delta R_2), \tag{2.21}$$

$$\Delta y = \frac{1}{2\cos\theta}(\Delta R_1 + \Delta R_2). \tag{2.22}$$

Compute a new estimate of position:

$$\hat{X}_u = \hat{X}_u + \Delta x, \quad \hat{Y}_u = \hat{Y}_u + \Delta y. \tag{2.23}$$

Continue to compute $\theta, \Delta R_1$ and ΔR_2 from these equations with new values of $\hat{X}_u$ and $\hat{Y}_u$.

Iterate Eqs. 2.17–2.23 and stop when Δx and Δy become less than the desired accuracy.

2.2.3 Satellite Selection and Dilution of Precision (DOP)

Just as in a land-based system, better accuracy is obtained by using reference points well separated in space. For example, the range measurements made to four reference points clustered together will yield nearly equal values. Position calculations involve range differences, and where the ranges are nearly equal, small relative errors are greatly magnified in the difference. This effect, brought about as a result of satellite geometry, is known as DOP. This means that range errors that occur from other causes such as clock errors are also magnified by the geometric effect.

The observation equations in three dimensions for each satellite with known coordinates (x_i, y_i, z_i) and unknown user coordinates (X, Y, Z) are given by

$$Z_\rho^i = \rho_{\mathrm{r}}^i = \sqrt{(x_i - X)^2 + (y_i - Y)^2 + (z_i - Z)^2} + C_{\mathrm{b}}, \tag{2.24}$$

where ρ_r^i is the pseudorange to the ith satellite and C_{b} is the receiver clock.

These are nonlinear equations that can be linearized using Taylor series (see, e.g., Chapter 5 of Ref. 4). The satellite positions have been converted to east–north–up (ENU) from earth-centered, earth-fixed (ECEF) coordinates (see Appendix B).

Let the vector of ranges be $\mathrm{Z}_\rho = \mathbf{h}(\mathbf{x})$, a nonlinear function $\mathbf{h}(\mathbf{x})$ of the four-dimensional vector $\mathbf{x}$ representing user position and receiver clock bias and expand the left-hand side of this equation in a Taylor series about some nominal solution $\mathbf{x}^{\mathrm{nom}}$ for the unknown vector

$$x = [X, Y, Z, C_{\mathrm{b}}]^T \tag{2.25}$$

of variables

X = east component of the user's antenna location,
Y = north component of the user's antenna location,
Z = upward vertical component of the user's antenna location, and
C_{b} = receiver clock bias,

for which

$$\begin{aligned} Z_\rho &= \mathbf{h}(\mathbf{x}) = \mathbf{h}\left(\mathbf{x}^{\mathrm{nom}}\right) + \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}\bigg|_{\boldsymbol{x}=\boldsymbol{x}^{\mathrm{nom}}} \delta\mathbf{x} + \text{ HOT}, \\ \delta\mathbf{x} &= \mathbf{x} - \mathbf{x}^{\mathrm{nom}}, \quad \delta Z_\rho = \mathbf{h}(\mathbf{x}) - \mathbf{h}\left(\mathbf{x}^{\mathrm{nom}}\right), \end{aligned} \tag{2.26}$$

where HOT stands for "higher-order terms."

These equations become

$$\begin{aligned} \delta Z_\rho &= \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}\bigg|_{\boldsymbol{x}=\boldsymbol{x}^{\text{nom}}} \frac{\partial \mathbf{x}}{} = \mathrm{H}^{[1]}\delta\mathbf{x}, \\ \delta x &= X - X_{\text{nom}}, \quad \delta y = Y - Y_{\text{nom}}, \quad \delta z = Z - Z_{\text{nom}}, \end{aligned} \tag{2.27}$$

where $H^{[1]}$ is the first-order term in the Taylor series expansion:

$$\delta Z_\rho = \rho_r(X, Y, Z) - \rho_r(X_{\text{nom}}, Y_{\text{nom}}, Z_{\text{nom}}) \tag{2.28}$$

$$\approx \underbrace{\frac{\partial \rho_r}{\partial X}\bigg|_{X_{\text{nom}}, Y_{\text{nom}}, Z_{\text{nom}}}}_{H^{[1]}} \delta\mathbf{x} + v_\rho \tag{2.29}$$

for v_ρ = noise in receiver measurements. This vector equation can be written in scalar form where i is the satellite number as

$$\left.\begin{aligned} \frac{\partial \rho_{\text{r}}^i}{\partial X} &= \frac{-(x_i - X)}{\sqrt{(x_i - X)^2 + (y_i - Y)^2 + (z_i - Z)^2}}\bigg|_{X_{\text{nom}}, Y_{\text{nom}}, Z_{\text{nom}}} \\ &= \frac{-(x_i - X_{\text{nom}})}{\sqrt{(x_i - X_{\text{nom}})^2 + (y_i - Y_{\text{nom}})^2 + (z_i - Z_{\text{nom}})^2}} \\ \frac{\partial \rho_{\text{r}}^i}{\partial Y} &= \frac{-(y_i - Y_{\text{nom}})}{\sqrt{(x_i - X_{\text{nom}})^2 + (y_i - Y_{\text{nom}})^2 + (z_i - Z_{\text{nom}})^2}} \\ \frac{\partial \rho_{\text{r}}^i}{\partial Z} &= \frac{-(z_i - Z_{\text{nom}})}{\sqrt{(x_i - X_{\text{nom}})^2 + (y_i - Y_{\text{nom}})^2 + (z_i - Z_{\text{nom}})^2}} \end{aligned}\right\} \tag{2.30}$$

for $i = 1, 2, 3, 4$ (i.e., four satellites).

We can combine Eqs. 2.29 and 2.30 into the matrix equation with measurements as

$$\underbrace{\begin{bmatrix} \partial z_\rho^1 \\ \partial z_\rho^2 \\ \partial z_\rho^3 \\ \partial z_\rho^4 \end{bmatrix}}_{4\times 1} = \underbrace{\begin{bmatrix} \frac{\partial \rho_{\text{r}}^1}{\partial X} & \frac{\partial \rho_{\text{r}}^1}{\partial Y} & \frac{\partial \rho_{\text{r}}^1}{\partial Z} & 1 \\ \frac{\partial \rho_{\text{r}}^2}{\partial X} & \frac{\partial \rho_{\text{r}}^2}{\partial Y} & \frac{\partial \rho_{\text{r}}^2}{\partial Z} & 1 \\ \frac{\partial \rho_{\text{r}}^3}{\partial X} & \frac{\partial \rho_{\text{r}}^3}{\partial Y} & \frac{\partial \rho_{\text{r}}^3}{\partial Z} & 1 \\ \frac{\partial \rho_{\text{r}}^4}{\partial X} & \frac{\partial \rho_{\text{r}}^4}{\partial Y} & \frac{\partial \rho_{\text{r}}^4}{\partial Z} & 1 \end{bmatrix}}_{4\times 4} \underbrace{\begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ C_{\text{b}} \end{bmatrix}}_{4\times 1} + \underbrace{\begin{bmatrix} v_\rho^1 \\ v_\rho^2 \\ v_\rho^3 \\ v_\rho^4 \end{bmatrix}}_{4\times 1},$$

which we can write in symbolic form as

$$\overbrace{\delta Z_\rho}^{4\times1} = \overbrace{H^{[1]}}^{4\times4}\overbrace{\delta\mathbf{x}}^{4\times1} + \overbrace{v_k}^{4\times1} \tag{2.31}$$

(see Table 5.3 in Ref. 4).

To calculate $H^{[1]}$, one needs satellite positions and the nominal value of the user's position in ENU coordinate frames.

To calculate the geometric dilution of precision (GDOP) (approximately), we obtain

$$\overbrace{\delta Z_\rho}^{4\times1} = \overbrace{H^{[1]}}^{4\times1}\overbrace{\delta\mathbf{x}}^{4\times1}. \tag{2.32}$$

Known are δZ_ρ and $H^{[1]}$ from the pseudorange, satellite position, and nominal value of the user's position. The correction $\delta\mathbf{x}$ is the unknown vector.

If we premultiply both sides of Eq. 2.32 by $H^{[1]T}$ the result will be

$$H^{[1]T}\delta Z_\rho = \underbrace{\overbrace{H^{[1]T}}^{4\times4}\overbrace{H^{[1]}}^{4\times4}}_{4\times4}\delta\mathbf{x}. \tag{2.33}$$

Then, we premultiply Eq. 2.33 by $(H^{[1]T}\,H^{[1]})^{-1}$:

$$\delta\mathbf{x} = \left(H^{[1]T}H^{[1]}\right)^{-1}H^{[1]T}\delta Z_\rho. \tag{2.34}$$

If $\delta\mathbf{x}$ and δZ_ρ are assumed random with zero mean, the error covariance (E = expected value)

$$E\left\langle(\delta\mathbf{x})(\delta\mathbf{x})^T\right\rangle = E\left\langle\left(H^{[1]T}H^{[1]}\right)^{-1}H^{[1]T}\delta Z_\rho\left[\left(H^{[1]T}H^{[1]}\right)^{-1}H^{[1]T}\delta Z_\rho\right]^T\right\rangle \tag{2.35}$$

$$= \left(H^{[1]T}H^{[1]}\right)^{-1}H^{[1]T}\underbrace{E\left\langle\delta Z_\rho\delta Z_\rho^T\right\rangle H^{[1]}}\left(H^{[1]T}H^{[1]}\right)^{-1}. \tag{2.36}$$

The pseudorange measurement covariance is assumed uncorrelated satellite to satellite with variance σ^2:

$$E\left\langle\delta Z_\rho\delta Z_\rho^T\right\rangle = \sigma^2\mathbf{I}. \tag{2.37}$$

Substituting Eq. 2.37 into Eq. 2.36 gives

$$E\left[\delta\mathbf{x}(\delta\mathbf{x})^T\right] = \sigma^2\left(H^{[1]T}H^{[1]}\right)^{-1}\underbrace{\left(H^{[1]T}H^{[1]}\right)\left(H^{[1]T}H^{[1]}\right)^{-1}}_{\mathbf{I}} \tag{2.38}$$

$$= \sigma^2\left(H^{[1]T}H^{[1]}\right)^{-1}, \tag{2.39}$$

for

$$\overbrace{\delta\mathbf{x}}^{4\times 1} = \begin{bmatrix} \Delta E \\ \Delta N \\ \Delta U \\ C_b \end{bmatrix},$$

and

$$\begin{matrix} \Delta E = \text{east error} \\ \Delta N = \text{north error} \\ \Delta U = \text{up error} \end{matrix} \quad \begin{pmatrix} \text{locally} \\ \text{level} \\ \text{coordinate} \\ \text{frame} \end{pmatrix},$$

and the covariance matrix becomes

$$\underbrace{E\left\langle \delta\mathbf{x}(\delta\mathbf{x})^T \right\rangle}_{} = \begin{bmatrix} E\langle \Delta E^2 \rangle & E\langle \Delta E \Delta N \rangle & E\langle \Delta E \Delta U \rangle & E\langle \Delta E \Delta C_b \rangle \\ E\langle \Delta N \Delta E \rangle & E\langle \Delta N^2 \rangle & E\langle \Delta N \Delta U \rangle & E\langle \Delta N \Delta C_b \rangle \\ E\langle \Delta U \Delta E \rangle & E\langle \Delta U \Delta N \rangle & E\langle \Delta U^2 \rangle & E\langle \Delta U \Delta C_b \rangle \\ E\langle \Delta C_b \Delta E \rangle & E\langle \Delta C_b \Delta N \rangle & E\langle \Delta C_b \Delta U \rangle & E\langle C_b^2 \rangle \end{bmatrix}. \quad (2.40)$$

We are principally interested in the diagonal elements of

$$\left(H^{[1]T} H^{[1]}\right)^{-1} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} \quad (2.41)$$

that represent the dilution of precision (DOP) of range measurement error to the user solution error (see Fig. 2.4):

$$\begin{aligned} \text{Geometric DOP (GDOP)} &= \sqrt{A_{11} + A_{22} + A_{33} + A_{44}}, \\ \text{Position DOP (PDOP)} &= \sqrt{A_{11} + A_{22} + A_{33}}, \\ \text{Horizontal DOP (HDOP)} &= \sqrt{A_{11} + A_{22}}, \\ \text{Vertical DOP (VDOP)} &= \sqrt{A_{33}}, \\ \text{Time DOP (TDOP)} &= \sqrt{A_{44}}. \end{aligned}$$

Hence, all DOPs represent the sensitivities of user solution error to pseudo-range errors. Figure 2.4 illustrates the relationship between the various DOP terms.

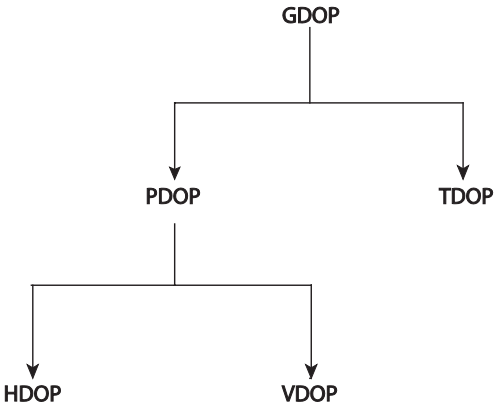


Fig. 2.4 DOP hierarchy.

2.2.4 Example Calculation of DOPS

2.2.4.1 Four Satellites For simplicity, consider four satellite measurements. The best accuracy is found with three satellites equally spaced on the horizon, at minimum elevation angle, with the fourth satellite directly overhead, as listed in Table 2.1.

TABLE 2.1. Example with Four Satellites

	Satellite Location			
	1	2	3	4
Elevation (deg)	5	5	5	90
Azimuth (deg)	0	120	240	0

The diagonal of the unscaled covariance matrix $(H^{[1]T}\,H^{[1]})^{-1}$ then has the terms

$$\begin{bmatrix} (\text{east DOP})^2 & & & \text{Cross-terms} \\ & (\text{north DOP})^2 & & \\ & & (\text{vertical DOP})^2 & \\ \text{Cross-terms} & & & (\text{time DOP})^2 \end{bmatrix},$$

where

$$\text{GDOP} = \sqrt{\text{trace}\left(H^{[1]T}H^{[1]}\right)^{-1}},\ H^{[1]} = \left.\frac{\partial \rho}{\partial x}\right|_{X_{\text{nom}},Y_{\text{nom}},Z_{\text{nom}}}.$$

Typical example values of $H^{[1]}$ for this geometry are

$$H^{[1]} = \begin{bmatrix} 0.000 & 0.996 & 0.087 & 1.000 \\ 0.863 & -0.498 & 0.087 & 1.000 \\ -0.863 & -0.498 & 0.087 & 1.000 \\ 0.000 & 0.000 & 1.000 & 1.000 \end{bmatrix}.$$

The GDOP calculations for this example are

$$\left(H^{[1]T} H^{[1]}\right)^{-1} = \begin{bmatrix} 0.672 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.672 & 0.000 & 0.000 \\ 0.000 & 0.000 & 1.600 & -.0505 \\ 0.000 & 0.000 & -0.505 & 0.409 \end{bmatrix},$$

$$\text{GDOP} = \sqrt{0.672 + 0.672 + 1.6 + .409} = 1.83$$

$$\text{PDOP} = 1.72$$

$$\text{HDOP} = 1.16$$

$$\text{VDOP} = 1.26$$

$$\text{TDOP} = 0.64.$$

2.3 TIME AND GPS

2.3.1 Coordinated Universal Time (UTC) Generation

UTC is the timescale based on the atomic second but is occasionally corrected by the insertion of leap seconds so as to keep it approximately synchronized with the earth's rotation. The leap second adjustments keep UTC within 0.9 s of UT1, which is a timescale based on the earth's axial spin. UT1 is a measure of the true angular orientation of the earth in space. Because the earth does not spin at exactly a constant rate, UT1 is not a uniform timescale [5].

2.3.2 GPS System Time

The timescale to which GPS signals are referenced is referred to as *GPS time*. GPS time is derived from a composite or "paper" clock that consists of all operational monitor station and satellite atomic clocks. Over the long run, it is steered to keep it within about 90 nanoseconds (1σ) of UTC, as maintained by the master clock at the U.S. Naval Observatory, ignoring the UTC leap seconds. At the integer second level, GPS time equaled UTC in 1980. However,

due to the leap seconds that have been inserted into UTC, GPS time was ahead of UTC by 16 s after June 2012.

2.3.3 Receiver Computation of UTC

The parameters needed to calculate UTC from GPS time are found in subframe 4 of the navigation data message. These data include a notice to the user regarding the scheduled future or recent past (relative to the navigation message upload) value of the delta time due to leap seconds Δt_{LFS}, together with the week number $\mathrm{WN}_{\mathrm{LFS}}$ and the day number DN, at the end of which the leap second becomes effective. The latter two quantities are known as the *effectivity time* of the leap second. "Day one" is defined as the first day relative to the end/start of a week and the $\mathrm{WN}_{\mathrm{LFS}}$ value consists of the eight least significant bits (LSBs) of the full week number.

Three different UTC/GPS time relationships exist, depending on the relationship of the effectivity time to the user's current GPS time:

1. *First Case.* Whenever the effectivity time indicated by the $\mathrm{WN}_{\mathrm{LFS}}$ and WN values is not in the past relative to the user's present GPS time, *and* the user's present time does not fall in the time span starting at DN +3/4 and ending at DN +5/4, the UTC time is calculated as

$$t_{\mathrm{UTC}} = t_E - \Delta t_{\mathrm{UTC}} \quad (\mathrm{modulo}\ 86{,}400)\mathrm{s},$$

where t_{UTC} is in seconds; 86,400 is the number of seconds per day; and

$$\Delta t_{\mathrm{UTC}} = \Delta T_{\mathrm{LS}} + A_0 + A_1\left[t_E - t_{0t} + 604{,}800(\mathrm{WN} - \mathrm{WN}_t)\right]\mathrm{s},$$

where 604,800 is the number of seconds per week, and

t_E = user GPS time from start of week (s),
ΔT_{LS} = delta time due to leap seconds,
A_0 = a constant polynomial term from the ephemeris message,
A_1 = a first-order polynomial term from the ephemeris message,
t_{0t} = reference time for UTC date,
WN = current week number derived from subframe 1,
WN_t = UTC reference week number.

The user GPS time t_E is in seconds relative to the end/start of the week, and the reference time t_{0t} for UTC data is referenced to the start of that week, whose number WN_t is given in word 8 of page 18 in subframe 4. The WN_t value consists of the eight LSBs of the full week number. Thus,

the user must account for the truncated nature of this parameter as well as truncation of WN, WN_t, and WN_{LFS} due to the rollover of the full week number. These parameters are managed by the GPS control segment so that the absolute value of the difference between the untruncated WN and WN_t values does not exceed 127.

2. *Second Case.* Whenever the user's current GPS time falls within the time span from DN +3/4 to DN +5/4, proper accommodation of the leap second event with a possible week number transition is provided by the following expression for UTC:

$$t_{UTC} = W[\text{modulo}(86,400 + \Delta t_{LSF} \Delta t_{LS})]\text{s},$$

where

$$W = (t_E - \Delta t_{UTC} - 43,200)(\text{modulo } 86,400) + 43,200 \text{ s},$$

and the definition of Δt_{UTC} given previously applies throughout the transition period.

3. *Third Case.* Whenever the effectivity time of the leap second event, as indicated by the $WN_{LFS}$and DN values, is in the past relative to the user's current GPS time, the expression given for t_{UTC} in the first case above is valid except that the value of Δt_{LFS} is used instead of Δt_{LS}. The GPS control segment coordinates the update of UTC parameters at a future upload in order to maintain a proper continuity of the t_{UTC} timescale.

2.4 EXAMPLE: USER POSITION CALCULATIONS WITH NO ERRORS

2.4.1 User Position Calculations

This section demonstrates how to go about calculating the user position, given ranges (pseudoranges) to satellites, the known positions of the satellites, and ignoring the effects of clock errors, receiver errors, propagation errors, and so on.

Then, the pseudoranges will be used to calculate the user's antenna location.

2.4.1.1 Position Calculations Neglecting clock errors, let us first determine the position calculation with no errors:

ρ_r = pseudorange (known),
x, y, z = satellite position coordinates (known), in ECEF,
X, Y, Z = user position coordinates (unknown),

where x, y, z, X, Y, Z are in the ECEF coordinate system. (It can be converted to ENU).

Position calculation with no errors is

$$\rho_{\mathrm{r}} = \sqrt{(x-X)^2 + (y-Y)^2 + (z-Z)^2}. \tag{2.42}$$

Squaring both sides yields

$$\rho_{\mathrm{r}}^2 = (x-X)^2 + (y-Y)^2 + (z-Z)^2 \tag{2.43}$$

$$= \underbrace{X^2 + Y^2 + Z^2}_{r^2 + C_{rr}} + x^2 + y^2 + z^2 - 2Xx - 2Yy - 2Zz, \tag{2.44}$$

$$\rho_{\mathrm{r}}^2 - \left(x^2 + y^2 + z^2\right) - r^2 = C_{rr} - 2Xx - 2Yy - 2Zz, \tag{2.45}$$

where r equals the radius of earth and C_{rr} is the clock bias correction. The four unknowns are (X, Y, Z, C_{rr}). Satellite position (x, y, z) is calculated from ephemeris data. For four satellites, Eq. 2.45 becomes

$$\begin{aligned}
\rho_{\mathrm{r}_1}^2 - \left(x_1^2 + y_1^2 + z_1^2\right) - r^2 &= C_{rr} - 2Xx_1 - 2Yy_1 - 2Zz_1, \\
\rho_{\mathrm{r}_2}^2 - \left(x_2^2 + y_2^2 + z_2^2\right) - r^2 &= C_{rr} - 2Xx_2 - 2Yy_2 - 2Zz_2, \\
\rho_{\mathrm{r}_3}^2 - \left(x_3^2 + y_3^2 + z_3^2\right) - r^2 &= C_{rr} - 2Xx_3 - 2Yy_3 - 2Zz_3, \\
\rho_{\mathrm{r}_4}^2 - \left(x_4^2 + y_4^2 + z_4^2\right) - r^2 &= C_{rr} - 2Xx_4 - 2Yy_4 - 2Zz_4,
\end{aligned} \tag{2.46}$$

with unknown 4×1 state vector

$$\begin{bmatrix} X \\ Y \\ Z \\ C_{rr} \end{bmatrix}.$$

We can rewrite the four equations in matrix form as

$$\begin{bmatrix} \rho_{\mathrm{r}_1}^2 - \left(x_1^2 + y_1^2 + z_1^2\right) - r^2 \\ \rho_{\mathrm{r}_2}^2 - \left(x_2^2 + y_2^2 + z_2^2\right) - r^2 \\ \rho_{\mathrm{r}_3}^2 - \left(x_3^2 + y_3^2 + z_3^2\right) - r^2 \\ \rho_{\mathrm{r}_4}^2 - \left(x_4^2 + y_4^2 + z_4^2\right) - r^2 \end{bmatrix} = \begin{bmatrix} -2x_1 - 2y_1 - 2z_1 1 \\ -2x_2 - 2y_2 - 2z_2 1 \\ -2x_3 - 2y_3 - 2z_3 1 \\ -2x_4 - 2y_4 - 2z_4 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ C_{rr} \end{bmatrix}$$

or

$$\overset{4\times1}{\widetilde{Y}} = \overset{4\times4}{\widetilde{M}}\,\overset{4\times1}{\widetilde{X}_{\rho}}, \tag{2.47}$$

where

Y = vector (known),
M = matrix (known),
X_ρ = vector (unknown).

Then, we premultiply both sides of Eq. 2.47 by M^{-1}:

$$\begin{aligned} M^{-1}Y &= M^{-1}M\mathrm{X}_\rho \\ &= \mathrm{X}_\rho \\ &= \begin{bmatrix} X \\ Y \\ Z \\ C_{rr} \end{bmatrix}. \end{aligned}$$

If the rank of M, the number of linear independent columns of the matrix M, is less than 4, then M will not be invertible.

2.4.2 User Velocity Calculations

The governing equation in this case is

$$\dot{\rho}_\mathrm{r} = \frac{(x-X)(\dot{x}-\dot{X})+(y-Y)(\dot{y}-\dot{Y})+(z-Z)(\dot{z}-\dot{Z})}{\rho_\mathrm{r}}, \tag{2.48}$$

where

$\dot{\rho}_\mathrm{r}$ = range rate (known),
ρ_r = range (known),
(x, y, z) = satellite positions (known),
$(\dot{x}, \dot{y}, \dot{z})$ = satelite rates (known),
X, Y, Z = user position (known from position calculations),
$(\dot{X}, \dot{Y}, \dot{Z})$ = user velocity (unknown),

and from Eq. 2.48,

$$-\dot{\rho}_\mathrm{r} + \frac{1}{\rho_\mathrm{r}}[\dot{x}(x-X)+\dot{y}(y-Y)+\dot{z}(z-Z)] = \left(\frac{x-X}{\rho_\mathrm{r}}\dot{X} + \frac{y-Y}{\rho_\mathrm{r}}\dot{Y} + \frac{z-Z}{\rho_\mathrm{r}}\dot{Z}\right). \tag{2.49}$$

For three satellites, Eq. 2.49 becomes

$$\begin{bmatrix} -\dot{\rho}_{r_1} + \dfrac{1}{\rho_{r_1}}[\dot{x}_1(x_1 - X) + \dot{y}_1(y_1 - Y) + \dot{z}_1(z_1 - Z)] \\ -\dot{\rho}_{r_2} + \dfrac{1}{\rho_{r_2}}[\dot{x}_2(x_2 - X) + \dot{y}_2(y_2 - Y) + \dot{z}_2(z_2 - Z)] \\ -\dot{\rho}_{r_3} + \dfrac{1}{\rho_{r_3}}[\dot{x}_3(x_3 - X) + \dot{y}_3(y_3 - Y) + \dot{z}_3(z_3 - Z)] \end{bmatrix} = \begin{bmatrix} \dfrac{(x_1 - X)}{\rho_{r_1}} & \dfrac{(y_1 - Y)}{\rho_{r_1}} & \dfrac{(z_1 - Z)}{\rho_{r_1}} \\ \dfrac{(x_2 - X)}{\rho_{r_2}} & \dfrac{(y_2 - Y)}{\rho_{r_2}} & \dfrac{(z_2 - Z)}{\rho_{r_2}} \\ \dfrac{(x_3 - X)}{\rho_{r_3}} & \dfrac{(y_3 - Y)}{\rho_{r_3}} & \dfrac{(z_3 - Z)}{\rho_{r_3}} \end{bmatrix} \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}. \tag{2.50}$$

Equation 2.50 becomes, where

D = known vector,
N = known matrix,
U_v = unknown user velocity vector,

$$\overbrace{D}^{3\times1} = \overbrace{N}^{3\times3}\overbrace{U_v}^{3\times1}, \tag{2.51}$$

$$\overbrace{U_v}^{3\times1} = N^{-1}D. \tag{2.52}$$

However, if the rank of N is <3, N will not be invertible.

PROBLEMS

Refer to Appendix B for coordinate system definitions and to Appendix B Section 3.10 for satellite orbit equations.

2.1 Which of the following coordinate systems is not rotating?

(a) North–east–down (NED)

(b) ENU

(c) ECEF

(d) Earth-centered inertial (ECI)

(e) Moon-centered, moon fixed

2.2 For the following GPS satellites, find the satellite position in ECEF coordinates at $t = 3s$. (*Hint*: See Appendix B.) Ω_0 and θ_0 are given below at time $t_0 = 0$:

	Ω_0 (deg)	θ_0 (deg)
(a)	326	68
(b)	26	34

2.3 Using the results of the previous problem, find the satellite positions in the local reference frame. Reference should be to the COMSAT facility in Santa Paula, California, located at 32.4° latitude, –119.2° longitude. Use coordinate shift matrix $S = 0$. (Refer to Appendix B, Section B.3.10.)

2.4 Given the following GPS satellite coordinates and pseudoranges:

	Ω_0 (deg)	θ_0 (deg)	ρ(m)
Satellite 1	326	68	2.324×10^7
Satellite 2	26	340	2.0755×10^7
Satellite 3	146	198	2.1103×10^7
Satellite 4	86	271	2.3491×10^7

(a) Find the user's antenna position in ECEF coordinates.

(b) Find the user's antenna position in locally level coordinates referenced to 0° latitude, 0° longitude. Coordinate shift matrix $S = 0$.

(c) Find the various DOPs.

2.5 Given two satellites in north and east coordinates

$$x(1) = 6.1464 \times 10^6,\ y(1) = 2.0172 \times 10^7 \text{ in meters,}$$
$$x(2) = 6.2579 \times 10^6,\ y(2) = -7.4412 \times 10^6 \text{ in meters,}$$

with pseudoranges

$$c\Delta t(1) = \rho_r(1) = 2.324 \times 10^7 \text{ in meters,}$$
$$c\Delta t(2) = \rho_r(2) = 2.0755 \times 10^7 \text{ in meters,}$$

and starting with an initial guess of x_{est}, y_{est}, find the user's antenna position.

2.6 Rank VDOP, HDOP, and PDOP from smallest (best) to largest (worst) under normal conditions:

(a) VDOP≤HDOP≤PDOP

(b) VDOP≤PDOP≤HDOP

(c) HDOP≤VDOP≤PDOP

(d) HDOP≤PDOP≤VDOP

(e) PDOP≤HDOP≤VDOP

(f) PDOP≤VDOP≤HDOP.

2.7 UTC time and the GPS time are offset by an integer number of seconds (e.g., 16 s as of June 2012) as well as a fraction of a second. The fractional part is approximately

(a) 0.1–0.5 s

(b) 1–2 ms

(c) 100–200 ns

(d) 10–20 ns

2.8 Show that $C_{\text{ENU}}^{\text{ECEF}} \times C_{\text{ECEF}}^{\text{ENU}} = I$, the 3×3 identity matrix. (*Hint*: $C_{\text{ENU}}^{\text{ECEF}} = \left[C_{\text{ECEF}}^{\text{ENU}}\right]^T$.)

2.9 A satellite position at time $t = 0$ is specified by its orbital parameters as $\Omega_0 = 92.847°$, $\theta_0 = 135.226°$, $\alpha = 55°$, $R = 26{,}560{,}000$ m.

(a) Find the satellite position at = 1 s, in ECEF coordinates.

(b) Convert the satellite position from (a) with user at

$$\begin{bmatrix} X_u \\ Y_u \\ Z_u \end{bmatrix}_{\text{ECEF}} = \begin{bmatrix} -2.430601 \\ -4.702442 \\ 3.546587 \end{bmatrix} \times 10^6 \text{ m}$$

from WGS84 (ECEF) to ENU coordinates with origin at

$$\theta = \text{local reference longitude} = 32.4°$$

$$\phi = \text{local reference latitude} = -119.2°.$$

REFERENCES

[1] Department of Transportation. *LORAN-C User's Handbook*, Department of Transportation, U.S. Coast Guard, Commandant Instruction M12562.3, Washington, DC, May 1990.

[2] T. Logsdon, *The NAVSTAR Global Positioning System*. Van Nostrand Reinhold, New York, 1992.

[3] M. S. Grewal and A. P. Andrews, Application of Kalman Filtering to GPS, INS, & Navigation, Short Course Notes, Kalman Filtering Consultant Associates, Anaheim, CA, January 2013.

[4] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB®*, 3rd ed. John Wiley & Sons, New York, 2008.

[5] D. W. Allan, N. Ashby, and C. C. Hodge, The Science of Timekeeping, Hewlett-Packard Application Note 1289, Palo Alto, CA, 1997.

3

FUNDAMENTALS OF INERTIAL NAVIGATION

An inertial system does for geometry . . . what a watch does for time.[1]
—Charles Stark Draper

In Draper's analogy quoted above, watches keep track of time by being set to the correct time then incrementing that time according to the inputs from a "time sensor" (a frequency source) to update that initial value.

Inertial systems do something similar, only with different variables—and they increment doubly. They need to be set to the correct position and velocity. Thereafter, they use measured accelerations to increment that initial velocity and use the resulting velocities to increment position.

3.1 CHAPTER FOCUS

The overview of inertial navigation in Section 1.3 included some of the history of the technology and examples of the more popular sensor designs.

The focus here is on how these sensors are integrated into a navigation system, including the following:

[1]Quoted by author Tom Pickens in "Doc Gyro and His Wonderful 'Where Am I?' Machine," *American Way Magazine*, 1972.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

1. terminology for the phenomenology and apparatus of inertial navigation
2. mathematical models for calibrating and compensating sensor errors to improve accuracy
3. methods for determining the different calibration parameters used
4. models and methods used for calculating unsensed gravitational accelerations
5. methods for determining initial conditions of attitude, velocity, and position
6. methods for integrating sensed attitude rates and accelerations
7. computer requirements for implementing these methods.

How this all affects navigation performance is covered in Chapter 11.

3.2 BASIC TERMINOLOGY

The following is a breakdown of some terminolgy used throughout the book. An expanded standardized terminology for inertial sensors may be found in Ref. 4, and that for inertial systems in Ref. 6.

Inertia is the propensity of bodies to maintain constant translational and rotational velocity, unless disturbed by forces or torques, respectively (Newton's first law or motion).

Inertial reference frames are coordinate frames in which Newton's laws of motion are valid. They cannot be rotating or accelerating. They are not necessarily the same as the **navigation coordinates**, which are typically dictated by the navigation problem at hand. Our problem is that we live in a rotating and accelerating environment here on Earth, and that defines the coordinate system we are already familiar with. "Locally level" coordinates used for navigation near the surface of the earth are rotating (with the earth) and accelerating (to counter gravity). Such rotations and accelerations must be taken into account in the practical implementation of inertial navigation.

Inertial sensors measure inertial accelerations and rotations, both of which are vector-valued variables.

Accelerometers are sensors for measuring inertial acceleration, also called **specific force** to distinguish it from what we call "gravitational acceleration." The point is, **accelerometers do not measure gravitational acceleration**. What accelerometers measure is modeled by Newton's second law as $a = F/m$, where F is the physically applied force (not including gravity) and m is the mass it is applied to. The force per unit mass, F/m, is called **specific force**, and accelerometers are sometimes called **specific force receivers**.

Gyroscopes (usually shortened to "gyros") are sensors for measuring rotation.

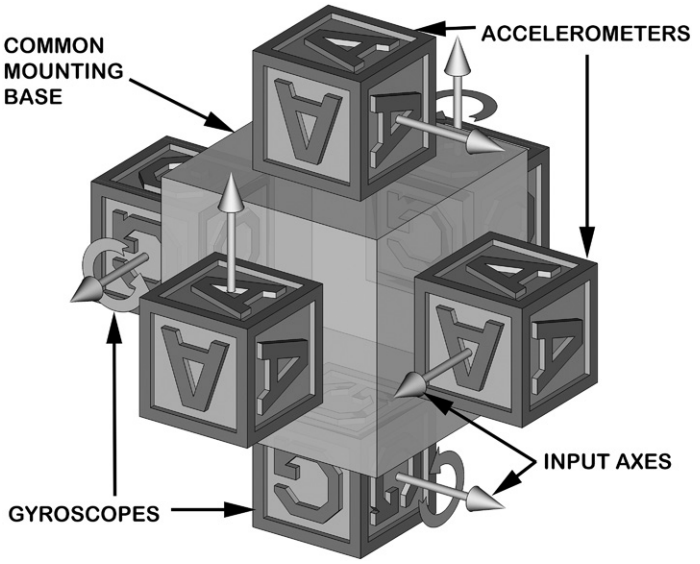


Fig. 3.1 Inertial sensor assembly (ISA) components.

Rate gyros measure rotation rates.

Displacement gyros (also called **whole-angle gyros**) measure accumulated rotation angles. Inertial navigation depends on gyros for maintaining knowledge of how the accelerometers are oriented in inertial and navigational coordinates.

Input axes of an inertial sensor define which vector components of acceleration or rotation rate it measures. These are illustrated by the arrows in Fig. 3.1, with rotation arrows wrapped around the input axes of gyroscopes to indicate the direction of rotation. Multiaxis sensors measure more than one component.

Calibration is a process for characterizing sensor behavior by observing input/output pairs, usually for the purpose of compensating sensor outputs to determine the sensor inputs.

Inertial sensor assemblies (ISAs) are ensembles of inertial sensors rigidly mounted to a common base to maintain the same relative orientations, as illustrated in Fig. 3.1. ISAs used in inertial navigation usually contain three accelerometers and three gyroscopes, represented in the figure by lettered blocks with arrows representing their respective input axes, or an equivalent configuration using multiaxis sensors. However, ISAs used for some other purposes (e.g., dynamic control applications such as autopilots or automotive steering augmentation) may not need as many sensors, and some designs use redundant sensors. Other terms used for the ISA are **instrument cluster** and (for gimbaled systems) **stable element** or **stable platform**.

Inertial reference unit (IRU) is a term commonly used for an inertial sensor system for attitude information only (i.e., using only gyroscopes). Space-based telescopes, for example, do not generally need accelerometers, but they do need gyroscopes to keep track of orientation.

Inertial measurement units (IMUs) include ISAs and associated support electronics for calibration and control of the ISA. Support electronics may also include thermal control or compensation, signal conditioning, and input/output control. An IMU may also include an IMU processor, and—for gimbaled systems—the gimbal control electronics.

Inertial navigation systems (INSs) measure rotation rates and accelerations, and calculate attitude, velocity, and position. Its subsystems include

IMUs, already mentioned above.

Navigation computers (one or more) to calculate the gravitational acceleration (not measured by accelerometers) and process the outputs of the accelerometers and gyroscopes from the IMU to maintain an estimate of the position of the IMU. Intermediate results of the implementation method usually include estimates of velocity, attitude, and attitude rates of the IMU.

User interfaces, such as display consoles for human operators and analog and/or digital data interfaces for vehicle guidance and control functions.

Power supplies and/or raw power conditioning for the complete INS.

Implementations of INSs include two general types:

Gimbaled systems use their gyroscopes for controlling ISA attitude. Commonly used ISA orientations include

Inertially stable (nonrotating), a common orientation for operations in space. In this case, the ISA may include one or more star trackers to correct for any gyroscope errors. However, locally level implementations may also use star trackers for the same purpose.

Locally level, a common orientation for terrestrial navigation. In this case, the ISA rotates with the earth and keeps two of its reference axes locally level during horizontal motion over the surface. Some early systems aligned the gyro and accelerometer input axes with the local directions of north, east, and down, because the gimbal angles could then represent the Euler angles for heading (yaw), pitch, and roll of the vehicle. However, there are also advantages in allowing the locally stabilized element to physically rotate about the local vertical direction.

Strapdown systems do nothing to physically control the orientation of the ISA, but they do process the gyroscope outputs to keep track of its orientation.

Gimbaled systems are generally more expensive than strapdown systems, but their performance is usually better. This is due, in part, to the fact that their gyroscopes and accelerometers are not required to endure high rotation rates.

Both gimbaled and strapdown systems commonly use some form of **shock and vibration isolation** to keep mechanical disturbances within the host vehicle frame from harming the sensors or the navigation implementation.

Because INSs perform integrals of acceleration and attitude rates, these integrals need initial values.

Alignment is a procedure for establishing the initial ISA attitude with respect to navigation coordinates. This can be done using external optical reference directions. However, systems with sufficiently accurate sensors can perform **self-alignment** when the system is stationary with respect to the earth. In that case, the implementation can be divided into two parts:

Leveling The implementation uses the accelerometers to measure the upward acceleration required to counter gravity, from which the system can determine the orientation of its ISA relative to local vertical. For gimbaled systems, the stable element (ISA) is physically leveled during this process (hence the name).

Gyrocompassing A procedure for estimating the direction of the earth's rotation axes with respect to ISA coordinates, using its gyroscopes. This and the direction of the local vertical then determine the north–south direction, so long as the stationary location is not in the vicinity of the poles. Given these two directions, the INS can orient itself relative to its location on the earth. The term *gyrocompassing* is a reference to the gyrocompass, an instrument introduced toward the end of the nineteenth century to replace the magnetic compass on iron ships. The gyrocompass uses only mechanical means to orient itself relative to north, whereas the INS requires a computer. For some gimbaled systems, gyrocompassing physically aligns the ISA with its level sensor axes pointing north and east.

Initialization is a procedure for establishing the initial position and velocity of the INS. Some of this can be done autonomously when the system is stationary with respect to the earth, in which case its relative velocity is zero in earth-fixed coordinates. The angle between the local vertical and the measured rotation axis of the earth, determined during alignment, can be used to estimate latitude. However, longitude and altitude must be determined by other means.

Host vehicle is a term used for the moving platforms on or in which an INS is mounted. It could be a spacecraft, aircraft, surface ship, submarine, land vehicle, or pack animal (including humans).

3.3 INERTIAL SENSOR ERROR MODELS

Inertial navigation has been called "navigation in a box" and "black-box navigation" because it is entirely self-contained. It infers what is going on outside by what it can sense inside.

Inertial sensors are also "black boxes" for the same reason. The problem is, there may be more going on outside the sensor than just accelerations and rotations,[2] as illustrated in Fig. 3.2. For the purpose of inertial navigation, one needs to know the sensor inputs, given the outputs. That is the art of inertial sensor modeling.

Mathematical models for how inertial sensors perform are used throughout the INS development cycle including the following:

1. In designing sensors to meet specified performance metrics.
2. To calibrate and compensate for fixed errors, such as scale factor and output bias. The extreme performance requirements for inertial sensors may not be attainable within manufacturing tolerances. Fortunately, the last few orders-of-magnitude improvement in performance can often be achieved through calibration. These calibration models are generally of three types:
   (a) Models based on engineering data and the principles of physics, such as the models carried over from the design trade-offs. These models generally have a known set of possible causes for each observed effect.

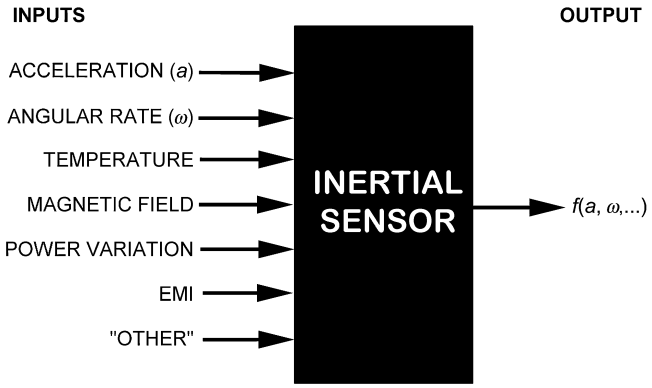


Fig. 3.2 Sensor black-box model.

[2]A comment often heard from inertial sensor designers is "No matter what sort of sensor we design, it always turns out to be a highly sensitive thermometer!"

(b) Abstract, general-purpose mathematical models such as polynomials, used to fit observed error data in such a way that the sensor output errors can be effectively corrected.

(c) Models for unpredictable variations in sensor output, used for predicting sensor and system performance.

3. Additional error models used in global navigation satellite system (GNSS)/INS integration for determining the optimal weighting (Kalman gain) in combining GNSS and INS navigation data.
4. Sensor models used in GNSS/INS integration for recalibrating the INS continuously while GNSS data are available. This approach gives the INS improved initial accuracy during periods of GNSS signal outage.

3.3.1 Zero-Mean Random Errors

These are the standard types of error models from Kalman filtering, used for modeling unpredictable outputs and described in Chapter 10.

3.3.1.1 White Sensor Noise This is usually lumped together under "electronic noise," which may come from power supplies, intrinsic noise in semiconductor devices, or from quantization errors in digitization.

3.3.1.2 Exponentially Correlated Noise Temperature sensitivity of sensor bias will often look like a time-varying additive noise source, driven by external ambient temperature variations or by internal heat distribution variations.

3.3.1.3 Random Walk Sensor Errors Random walk errors are characterized by variances that grow linearly with time and power spectral densities that fall off as 1/frequency^2 (i.e., 20 dB per decade).

There are specifications for random walk noise in inertial sensors, but mostly for the integrals of their outputs and not in the outputs themselves. For example, the "angle random walk" from a rate gyroscope is equivalent to white noise in the angular rate outputs. In a similar fashion, the integral of white noise in accelerometer outputs would be equivalent to a "velocity random walk."

The random walk error model has the form

$$\varepsilon_k = \varepsilon_{k-1} + w_{k-1}$$

$$\begin{aligned}\sigma_k^2 &\overset{\text{def}}{=} \left\langle \varepsilon_k^2 \right\rangle \\ &= \sigma_{k-1}^2 + \left\langle w_{k-1}^2 \right\rangle \\ &= \sigma_0^2 + kQ_w \text{ for time-invariant systems}\end{aligned}$$

$$Q_w \overset{\text{def}}{=} \underset{k}{E}\left\langle w_k^2 \right\rangle.$$

The value of Q_w will be in units of squared error per discrete time step Δt. Random walk error sources are usually specified in terms of standard deviations, that is, error units per square root of time unit. Gyroscope angle random walk errors, for example, might be specified in $\text{deg}/\sqrt{\text{h}}$. Most navigation-grade gyroscopes (including RLG, HRG, IFOG) have angle random walk errors in the order of 10^{-3} $\text{deg}/\sqrt{\text{h}}$ or less.

3.3.1.4 Harmonic Noise Temperature control schemes (including building HVAC systems) often introduce cyclical errors due to thermal transport lags, and these can cause harmonic errors in sensor outputs, with harmonic periods that scale with device dimensions. Also, suspension and structural resonances of host vehicles introduce harmonic accelerations, which can excite acceleration-sensitive error sources in sensors.

3.3.1.5 "1/f" Noise This noise is characterized by power spectral densities that fall off as $1/f$, where f is the frequency. It is present in most electronic devices, its causes are not well understood, and it is usually modeled as some combination of white noise and random walk.

3.3.2 Fixed-Pattern Errors

These are repeatable sensor output errors, unlike the zero-mean random noise considered above. The same types of models apply to accelerometers and gyroscopes. Some of the more common types of sensor errors are illustrated in Fig. 3.3. These are

(a) bias, which is any nonzero sensor output when the input is zero;
(b) scale factor error, usually due to manufacturing tolerances;
(c) nonlinearity, which is present in most sensors to some degree;
(d) scale factor sign asymmetry (often from mismatched push–pull amplifiers);
(e) a dead zone, usually due to mechanical stiction or lock-in (for ring laser gyroscopes); and
(f) quantization error, inherent in all digitized systems; it may not be zero-mean when the input is held constant, as it could be under calibration conditions.

We can recover the sensor input from the sensor output so long as the input/output relationship is known and invertible. Dead-zone errors and quantization errors are the only ones shown with this problem. The cumulative effects of both types (dead zone and quantization) often benefit from zero-mean input noise or dithering. Also, not all digitization methods have equal cumulative effects. Cumulative quantization errors for sensors with frequency outputs are bounded by ± one-half least significant bit (LSB) of the digitized output,

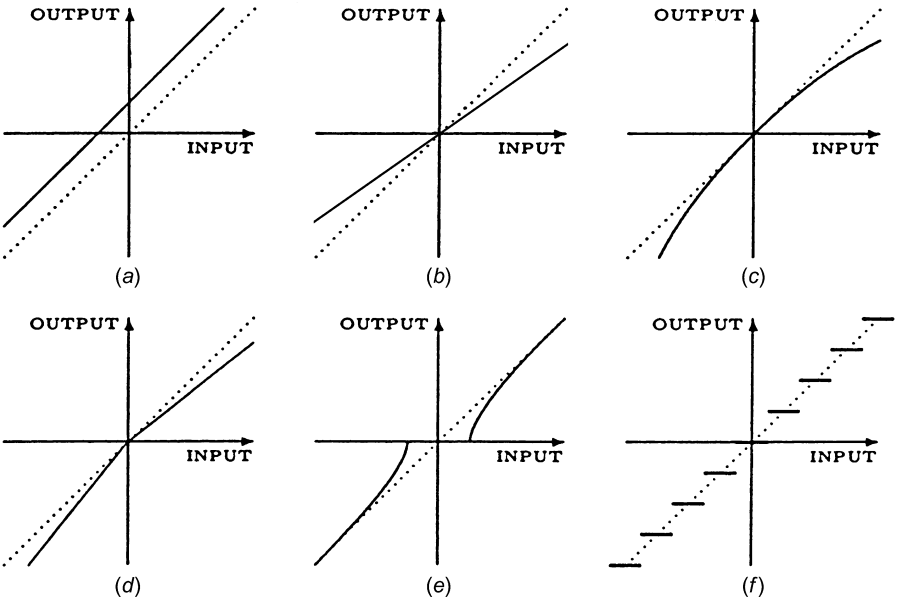


Fig. 3.3 Common input/output error types. (*a*) Bias. (*b*) Scale factor. (*c*) Nonlinearity. (*d*) ±Assymetry. (*e*) Dead zone. (*f*) Quantization.

but the variance of cumulative errors from independent sample-to-sample A/D conversion errors can grow linearly with time.

3.3.3 Sensor Error Stability

Outright sensor failure is a serious problem in inertial navigation, and sensor reliability is always addressed in system design. Also, much attention has been given to detecting and correcting sensor failures during operation. Slow sensor accuracy degradation is another problem.

In practice, fixed-pattern sensor errors do not necessarily remain fixed over long periods of time (hours to years). Some of this may be due to second-order sensitivities to ambient conditions (e.g., temperature, barometric pressure, humidity, power levels, and magnetic fields) or may be attributed to "aging." Truly second-order effects can be calibrated and compensated, but compensation requires additional sensors for the second variable, and calibration adds cost.

Navigation errors due to sensor instability can be compensated to some degree by integrating INS with other sensor systems, including GNSS. This requires models for the expected patterns of sensor degradation, a subject addressed in the next section.

3.4 SENSOR CALIBRATION AND COMPENSATION

Sensor compensation is the process of recovering the sensor inputs from the sensor outputs.

Sensor calibration is the process of determining the parameters of the compensation model.

3.4.1 Sensor Biases, Scale Factors, and Misalignments

This part of sensor compensation can be done using an affine (linear plus offset) model. The biases are offsets and the rest is linear.

3.4.1.1 Compensation Model Parameters The model used here is for ISA-level calibration. Calibration can also be done at the sensor level, but it is less expensive if it is done at the ISA level.

The changes in sensor input/output patterns due to biases and scale factors are illustrated in Fig. 3.3. Figure 3.4 illustrates how input axis misalignments and scale factors at the ISA level affect sensor ouputs, in terms of how they are related to the linear input/output model:

$$\mathbf{z}_{\text{output}} = \mathbf{M}\left(\mathbf{z}_{\text{input}} + \mathbf{b}_z\right) \tag{3.1}$$

$$\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}, \tag{3.2}$$

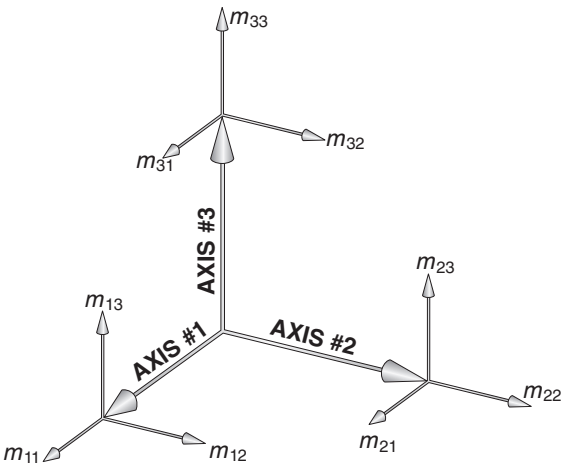


Fig. 3.4 Directions of modeled sensor cluster errors.

where $\mathbf{z}_{\text{input}}$ is a vector representing the inputs (accelerations or rotation rates) to three inertial sensors with nominally orthogonal input axes, $\mathbf{z}_{\text{output}}$ is a vector representing the corresponding outputs, $\mathbf{b}_z$ is a vector of sensor output biases, and the corresponding elements of $\mathbf{M}$ are labeled in Fig. 3.4.

The parameters m_{ij} and $\mathbf{b}_z$ of this model can be estimated from observations of sensor outputs when the inputs are known, the process called **calibration**.

The purpose of calibration is **sensor compensation**, which can be accomplished by inverting the "forward model" of Eq. 3.1 to obtain

$$\mathbf{z}_{\text{input}} = \mathbf{M}^{-1}\mathbf{z}_{\text{output}} - \mathbf{b}_z, \tag{3.3}$$

the sensor inputs compensated for scale factor, misalignment, and bias errors.

This result can be generalized for a cluster of $N \geq 3$ gyroscopes or accelerometers; the effects of individual **biases**, **scale factors**, and **input axis misalignments** can be modeled by an equation of the form

$$\underbrace{\mathbf{z}_{\text{input}}}_{3\times 1} = \underbrace{\mathbf{M}^{\dagger}_{\text{scale factor \& misalignment}}}_{3\times N}\underbrace{\mathbf{z}_{\text{output}}}_{N\times 1} - \underbrace{\mathbf{b}_z}_{3\times 1}, \tag{3.4}$$

where $\mathbf{M}^{\dagger}$ is the Moore–Penrose pseudoinverse of the corresponding $\mathbf{M}$, which can be determined by calibration.

3.4.1.2 Calibrating Sensor Biases, Scale Factors, and Misalignments In this case, calibration amounts to estimating the values of $\mathbf{M}^{\dagger}$ and $\mathbf{b}_z$, given input–output pairs $[\mathbf{z}_{\text{input},\,k}, \mathbf{z}_{\text{output},\,k}]$, where $\mathbf{z}_{\text{input},\,k}$ is known from controlled calibration conditions and $\mathbf{z}_{\text{output},\,k}$ is recorded under these conditions. For accelerometers, controlled conditions may include the direction and magnitude of gravity, conditions on a shake table, or those on a centrifuge. For gyroscopes, controlled conditions may include the relative direction of the rotation axis of Earth (e.g., with sensors mounted on a two-axis indexed rotary table), or controlled conditions on a rate table.

The full set of input/output pairs under K sets of calibration conditions yields a system of $3K$ linear equations

$$\underbrace{\begin{bmatrix} z_{1,\text{input},1} \\ z_{2,\text{input},1} \\ z_{3,\text{input},1} \\ \vdots \\ z_{3,\text{input},K} \end{bmatrix}}_{3K\text{ knowns}} = \underbrace{\begin{bmatrix} z_{1,\text{output},1} & z_{2,\text{output},1} & z_{3,\text{output},1} & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}}_{Z,\text{ a } 3K\times(3N+3)\text{ matrix of knowns}} \underbrace{\begin{bmatrix} m_{1,1} \\ m_{1,2} \\ m_{1,3} \\ \vdots \\ b_{3,z} \end{bmatrix}}_{3N+3\text{ unknowns}} \tag{3.5}$$

in the $3N$ unknown parameters $m_{i,j}$ (the elements of the matrix $\mathbf{M}^{\dagger}$) and 3 unknown parameters $b_{i,z}$ (rows of the 3-vector $\mathbf{b}_z$), which will be overdetermined for $K > N + 1$. In that case, the system of linear equations may be solv-

able for the $3(N+1)$ calibration parameters by using the method of least squares:

$$\begin{bmatrix} m_{1,1} \\ m_{1,2} \\ m_{1,3} \\ \vdots \\ b_{3,z} \end{bmatrix} = \left[\mathbf{Z}^{\mathbf{T}}\mathbf{Z}\right]^{-1}\mathbf{Z}^{\mathbf{T}} \begin{bmatrix} z_{1,\text{input},1} \\ z_{2,\text{input},1} \\ z_{3,\text{input},1} \\ \vdots \\ z_{3,\text{input},K} \end{bmatrix}, \tag{3.6}$$

provided that the matrix $\mathbf{Z}^{\mathbf{T}}\mathbf{Z}$ is nonsingular.

The values of $\mathbf{M}^{\dagger}$ and $\mathbf{b}_z$ determined in this way are called **calibration parameters**.

Estimation of the calibration parameters can also be done using Kalman filtering, a by-product of which would be the covariance matrix of calibration parameter uncertainty. This covariance matrix is also useful in modeling system-level performance.

3.4.2 Other Calibration Parameters

3.4.2.1 Nonlinearities Sensor input–output nonlinearities are generally modeled by polynomials:

$$z_{\text{input}} = \sum_{i=0}^{N} a_i z_{\text{output}}^{i}, \tag{3.7}$$

where the first two parameters a_0 = bias and a_1 = scale factor. The polynomial input–output model of Eq. 3.7 is linear in the calibration parameters, so they can still be calibrated using a system of linear equations—as was used for scale factor and bias.

The generalization of Eq. 3.7 to vector-valued inputs and outputs includes all the cross-power terms between different sensors, but it also includes multidimensional data structures in place of the scalar parameters a_i. Such a model would, for example, include the acceleration sensitivities of gyroscopes and the rotation rate sensitivities of accelerometers.

3.4.2.2 Sensitivities to Other Measurable Conditions Most inertial sensors are also thermometers, and part of the art of sensor design is to minimize their temperature sensitivities. Other bothersome sensitivities include acceleration sensitivity of gyroscopes and rotation rate sensitivities of accelerometers (already mentioned above).

Compensating for temperature sensitivity requires adding one or more thermometers to the sensors and taking calibration data over the expected

operational temperature range, but the other sensitivities can be "cross compensated" by using the outputs of the other inertial sensors. The accelerometer outputs can be used in compensating for acceleration sensitivities of gyroscopes, and the gyro outputs can be used in compensating for angular rate sensitivities of accelerometers.

3.4.2.3 Other Accelerometer Models *Centrifugal Acceleration Effects* Accelerometers have input axes defining the component(s) of acceleration that they measure. There is a not-uncommon superstition that these axes must intersect at a point to avoid some unspecified error source. That is generally not the case, but there can be some differential sensitivity to centrifugal accelerations due to high rotation rates and relative displacements between accelerometers. The effect is rather weak but not always negligible. It is modeled by the equation

$$a_{i,\text{centrifugal}} = \omega^2 r_i, \tag{3.8}$$

where ω is the rotation rate and r_i is the displacement component along the input axis from the axis of rotation to the effective center of the accelerometer. Even manned vehicles can rotate at $\omega \approx 3$ rad/s, which creates centrifugal accelerations of about $1\,g$ at $r_i = 1$ m and $0.001\,g$ at 1 mm. The problem is less significant, if not insignificant, for microelectromechanical system (MEMS)-scale accelerometers that can be mounted within millimeters of one another.

Center of Percussion Because ω can be measured, sensed centrifugal accelerations can be compensated, if necessary. This requires designating some reference point within the instrument cluster and measuring the radial distances and directions to the accelerometers from that reference point. The point within the accelerometer required for this calculation is sometimes called its "center of percussion." It is effectively the point such that rotations about all axes through the point produce no sensible centrifugal accelerations, and that point can be located by testing the accelerometer at differential reference locations on a rate table.

Angular Acceleration Sensitivities Pendulous accelerometers are sensitive to angular acceleration about their hinge lines, with errors equal to $\dot{\omega}\Delta_{\text{hinge}}$, where $\dot{\omega}$ is the angular acceleration in radian per second squared and Δ_{hinge} is the displacement of the accelerometer proof mass (at its center of mass) from the hinge line. This effect can reach the $1\,g$ level for $\Delta_{\text{hinge}} \approx 1$ cm and $\dot{\omega} \approx 10^3$ rad/s^2, but these extreme conditions are usually not persistent enough to matter in most applications.

3.4.3 Calibration Parameter Instabilities

INS calibration parameters are not always exactly constant. Their values can change over the operational life of the INS. Specifications for calibration

stability generally divide these calibration parameter variations into two categories: (1) changes from one system turn-on to the next and (2) slow "parameter drift" during operating periods.

3.4.3.1 Calibration Parameter Changes between Turn-Ons These are changes that occur between a system shutdown and the next start-up. They may be caused by temperature transients during shutdowns and turn-ons, or by what is termed "aging." They are generally considered to be independent from turn-on to turn-on, so the model for the covariance of calibration errors for the kth turn-on would be of the form

$$\mathbf{P}_{\text{calib.},k} = \mathbf{P}_{\text{calib.},k-1} + \Delta\mathbf{P}_{\text{calib.}}, \tag{3.9}$$

where $\Delta\mathbf{P}_{\text{calib.}}$ is the covariance of turn-on-to-turn-on parameter changes. The initial value $\mathbf{P}_{\text{calib.},0}$ at the end of calibration is usually determinable from error covariance analysis of the calibration process. Note that this is the covariance model for a random walk, the covariance of which grows without bound.

3.4.3.2 Calibration Parameter Drift This term applies to changes that occur in the operational periods between start-ups and shutdowns. The calibration parameter uncertainty covariance equation has the same form as Eq. 3.9, but with $\Delta\mathbf{P}_{\text{calib.}}$ now representing the calibration parameter drift in the time interval $\Delta t = t_k - t_{k-1}$ between successive discrete times within an operational period.

Detecting Error Trends Incipient sensor failures can sometimes be predicted by observing the variations over time of the sensor calibration parameters. One of the advantages of tightly coupled GNSS/INS integration is that INS sensors can be continuously calibrated all the time that GNSS data are available. System health monitoring can then include tests for the trends of sensor calibration parameters, setting threshold conditions for failing the INS system, and isolating a likely set of causes for the observed trends.

3.4.4 Auxilliary Sensors before GNSS

3.4.4.1 Attitude Sensors Nongyroscopic attitude sensors can also be used as aids in inertial navigation. These include the following:

Magnetic sensors are used primarily for coarse heading initialization to speed up INS alignment.

Star trackers are used primarily for space-based or near-space applications. The Snark cruise missile and the U-2 spy plane used inertial-platform-mounted star trackers to maintain INS alignment on long flights, an idea attributed to Northrop.

Optical alignment systems have been used on some systems prior to launch. Some use Porro prisms mounted on the inertial platform to maintain

TABLE 3.1. Performance Grades for Gyroscopes

Performance Parameter	Performance Units	Performance Grades		
		Inertial	Intermediate	Moderate
Max. input	deg/h	10^2–10^6	10^2–10^6	10^2–10^6
	deg/s	10^{-2}–10^2	10^{-2}–10^2	10^{-2}–10^2
Scale factor	part/part	10^{-6}–10^{-4}	10^{-4}–10^{-3}	10^{-3}–10^{-2}
Bias stability	deg/h	10^{-4}–10^{-2}	10^{-2}–10	10–10^2
	deg/s	10^{-8}–10^{-6}	10^{-6}–10^{-3}	10^{-3}–10^{-2}
Bias drift	deg / $\sqrt{h}$	10^{-4}–10^{-3}	10^{-2}–10^{-1}	1–10
	deg / $\sqrt{s}$	10^{-6}–10^{-5}	10^{-5}–10^{-4}	10^{-4}–10^{-3}

optical line-of-sight reference through ground-based theodolites to reference directions at the launch complex.

3.4.4.2 Altitude Sensors These include barometric altimeters and radar altimeters. Without GNSS inputs, some sort of altitude sensor is required to stabilize INS vertical channel errors.

3.4.5 Sensor Performance Ranges

Table 3.1 lists some order-of-magnitude performance ranges for gyroscopes, in terms of the stabilities of their error characteristics. The ranges are labeled

Inertial for those acceptable in stand-alone INS applications.

Intermediate for those acceptable for some applications of integrated GNSS/INS navigators.

Moderate for those considered acceptable only for low-grade integrated GNSS/INS navigators.

These are only rough order-of-magnitude ranges for the different error characteristics. Sensor requirements are largely determined by the application. For example, gyroscopes for gimbaled systems can generally use much smaller input ranges than those for strapdown applications.

The requirements for a specific application are best determined through systems analysis, described in Chapter 11.

3.5 EARTH MODELS

For navigation in the terrestrial environment, both inertial navigation and satellite navigation require models for the shape, gravity, and rotation of the earth.

Both systems use a common set of navigation coordinates fitted to a model of the shape of the earth, and this coordinate system rotates with the earth.

Gravity modeling is important for INS because gravity cannot be sensed. It is necessary for filling in the unsensed acceleration of terrestrial navigation coordinates, and it is necessary for GNSS in determining precise ephemerides of the satellites.

3.5.1 Terrestrial Navigation Coordinates

Descriptions of the major coordinates used in inertial navigation and GNSS/INS integration are described in Appendix B. These include coordinate systems used for representing the trajectories of GNSS satellites and user vehicles in the near-earth environment and for representing the attitudes of host vehicles relative to locally level coordinates, including the following:

1. Inertial coordinates:
   (a) Earth-centered inertial (ECI), with origin at the center of mass of the earth and principal axes in the directions of the vernal equinox and the rotation axis of the earth.
   (b) Satellite orbital coordinates, used in GNSS ephemerides.
2. Earth-fixed coordinates:
   (a) Earth-centered, earth-fixed (ECEF), with origin at the center of mass of the earth and principal axes in the directions of the prime meridian at the equator and the rotation axis of the earth.
   (b) Geodetic coordinates, based on an ellipsoid model for the shape of the earth. Longitude in geodetic coordinates is the same as in ECEF coordinates, and geodetic latitude as defined as the angle between the equatorial plane and the normal to the reference ellipsoid surface. Geodetic latitude can differ from geocentric latitude by as much as 12 arc minutes, equivalent to about 20 km of northing distance.
   (c) Local tangent plane (LTP) coordinates, also called "locally level coordinates," essentially representing the earth as being locally flat. These coordinates are particularly useful from a human factor standpoint for representing the attitude of the host vehicle and for representing local directions. They include
      i. east–north–up (ENU), shown in Fig. B.7;
      ii. north–east–down (NED), which can be simpler to relate to vehicle coordinates; and
      iii. alpha wander, rotated from ENU coordinates through an angle α about the local vertical, as shown in Fig. B.8 and described in Section 3.6.3.1.
3. Vehicle-fixed coordinates:
   (a) roll–pitch–yaw (RPY), as shown in Fig. B.9.

Transformations between these different coordinate systems are important for representing vehicle attitudes, for resolving inertial sensor outputs into inertial navigation coordinates, and for GNSS/INS integration. Methods used for representing and implementing coordinate transformations are also presented in Appendix B, Section B.4.

3.5.2 Earth Rotation

Earth is the mother of all clocks. It has given us the time units of days, hours, minutes, and seconds we use to manage our lives. Not until the discovery of atomic clocks based on hyperfine quantum state transitions were we able to observe the imperfections in our earth clock. Despite these, we continue to use earth rotation as our primary time reference, adding or subtracting leap seconds to atomic clocks to keep them synchronized to the rotation of the earth. These time variations are significant for GNSS navigation but not for inertial navigation.

World Geodetic System 1984 (WGS84) earthrate model: A **geoid** is a model for the equipotential surface of the earth at sea level, referenced to the center of mass of Earth and its rotation axis. "WGS84" refers to the 1984 World Geodetic Survey, a cooperative international effort undertaken to determine a more precise model of the shape of the earth. This effort had begun in the 1950s. Member nations shared data and developed a common geoid model. A series of such efforts resulted in a series of such models, each based on more data. The 1984 values have become standard for most applications.

The value of earthrate in the WGS84 earth model used by the Global Positioning System (GPS) is $7{,}292{,}115{,}167 \times 10^{-14}$ rad/s or about 15.04109 deg/h. This is its **sidereal** rotation rate with respect to distant stars. Its mean rotation rate with respect to the nearest star (our sun), as viewed from the rotating earth, is 15 deg/h, averaged over 1 year. We also know that Earth is slowing down with age due to its transfer of energy and angular momentum to the moon through the effects of tides.[3]

3.5.3 Gravity Models

Gravity may be part of Newtonian mechanics, but it is more appropriately modeled as a warping of the space–time continuum. The important point is **inertial sensors cannot measure gravity**.

Newton could observe the apple falling, but he could not feel gravity pulling the apple (or himself) downward. He could only feel the surface of the earth pushing him upward to counter it.

[3]An effect discovered by George Darwin (1845–1912), son of Charles Darwin.

That leads to one of the problems that had to be solved for inertial navigation in the terrestrial environment: How can we account for gravitational acceleration if we cannot measure it?

The solution is to do what Newton did: Model it.

3.5.3.1 GNSS Gravity Models Accurate gravity modeling is important for maintaining ephemerides for GNSS satellites, and models developed for GNSS have been a boon to inertial navigation as well. However, spatial resolution of the earth gravitational field required for GNSS operation may be a bit coarse compared to that for precision inertial navigation because the GNSS satellites are not near the surface and the mass concentration anomalies that create surface gravity anomalies. GNSS orbits have very little sensitivity to equipotential surface-level undulations of the gravitational field with wavelengths on the order of 100 km or less, but these can be important for high-precision inertial systems.

3.5.3.2 INS Gravity Models Because an INS operates in a world with gravitational accelerations it is unable to sense and unable to ignore, it must use a reasonably faithful model of gravity.

Gravity models for the earth include centrifugal acceleration due to the rotation of the earth as well as true gravitational accelerations due to the mass distribution of the earth, but they do not generally include oscillatory effects such as tidal variations.

Gravitational Potential Gravitational potential is defined to be zero at a point infinitely distant from all massive bodies and to decrease toward massive bodies such as the earth; that is, a point at infinity is the reference point for gravitational potential.

In effect, the gravitational potential at a point in or near the earth is defined by the potential energy lost per unit of mass falling to that point from infinite altitude. In falling from infinity, potential energy is converted to kinetic energy, $mv_{\text{escape}}^2 / 2$, where v_{escape} is the *escape velocity*. Escape velocity at the surface of the earth is about 11 km/s.

Gravitational Acceleration Gravitational acceleration is the negative gradient of gravitational potential. Potential is a scalar function, and its gradient is a vector. Because gravitational potential increases with altitude, its gradient points upward and the negative gradient points downward.

Equipotential Surfaces An equipotential surface is a surface of constant gravitational potential. If the ocean and atmosphere were not moving, then the surface of the ocean at static equilibrium would be an equipotential surface. **Mean sea level** is a theoretical equipotential surface obtained by time averaging the dynamic effects.

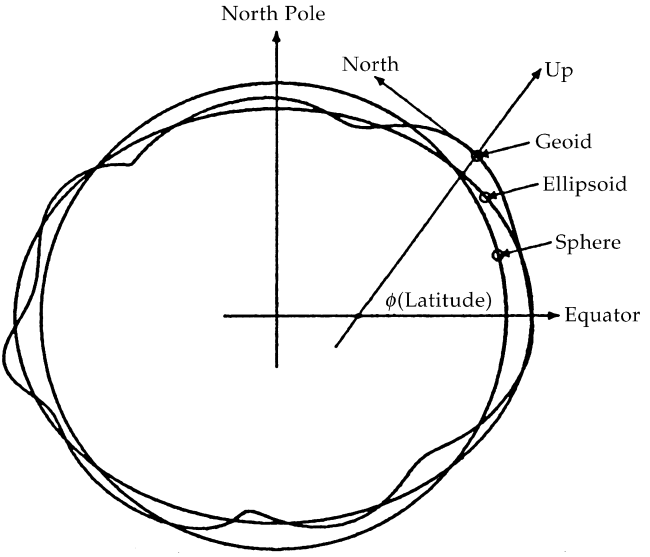


Fig. 3.5 Equipotential surface models for Earth.

Ellipsoid Models for Earth Geodesy is the process of determining the shape of the earth, often using ellipsoids as approximations of an equipotential surface (e.g., mean sea level), as illustrated in Fig. 3.5. The most common ones are ellipsoids of revolution, but there are many reference ellipsoids based on different survey data. Some are global approximations and some are local approximations. The global approximations deviate from a spherical surface by about ±10 km, and locations on the earth referenced to different ellipsoidal approximations can differ from one another by 10^2–10^3 m.

Geodetic latitude on a reference ellipsoid is measured in terms of the angle between the equator and the normal to the ellipsoid surface, as illustrated in Fig. 3.5.

Orthometric height is measured along the (curved) plumb line.

WGS84 Ellipsoid The WGS84 earth model approximates mean sea level (an equipotential surface) by an ellipsoid of revolution with its rotation axis coincident with the rotation axis of the earth, its center at the center of mass of the earth, and its prime meridian through Greenwich. Its semimajor axis (equatorial radius) is defined to be 6,378,137 m, and its semiminor axis (polar radius) is defined to be 6,356,752.3142 m.

Geoid Models Geoids are approximations of mean sea-level orthometric height with respect to a reference ellipsoid. Geoids are defined by additional higher-order shapes, commonly modeled by spherical harmonics of height deviations from an ellipsoid, as illustrated in Fig. 3.5. There are many geoid

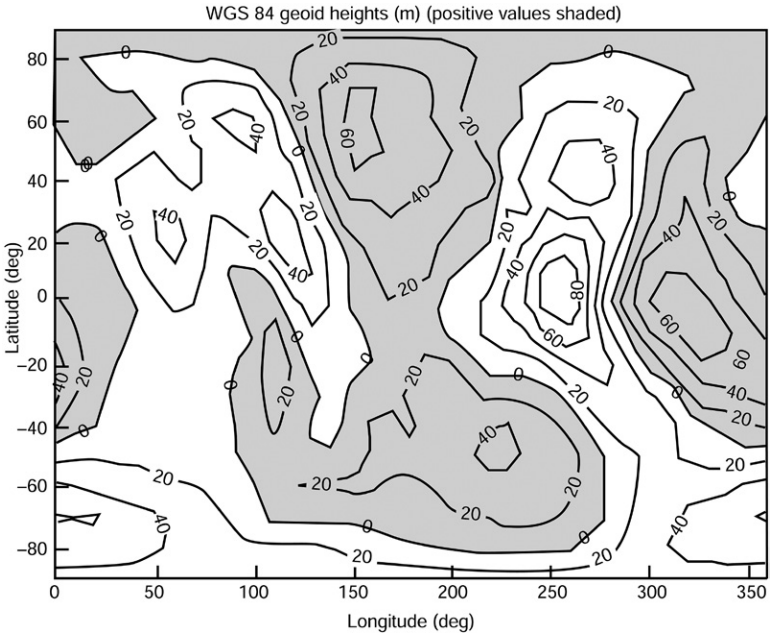


Fig. 3.6 WGS84 geoid heights.

models based on different data, but the more recent, most accurate models depend heavily on GPS data. Geoid heights deviate from reference ellipsoids by tens of meters, typically.

The WGS84 geoid heights vary about ±100 m from the reference ellipsoid. As a rule, oceans tend to have lower geoid heights and continents tend to have higher geoid heights. Coarse 20-m contour intervals are plotted versus longitude and latitude in Fig. 3.6, with geoid regions above the ellipsoid shaded gray.

3.5.3.3 Longitude and Latitude Rates The second integral of acceleration in locally level coordinates should result in the estimated vehicle position. This integral is somewhat less than straightforward when longitude and latitude are the preferred horizontal location variables.

The rate of change of vehicle altitude equals its vertical velocity, which is the first integral of net (i.e., including gravity) vertical acceleration. The rates of change of vehicle longitude and latitude depend on the horizontal components of vehicle velocity, but in a less direct manner. The relationship between longitude and latitude rates and east and north velocities is further complicated by the oblate shape of the earth.

The rates at which these angular coordinates change as the vehicle moves tangent to the surface will depend upon the radius of curvature of the reference surface model, which is an ellipsoid of revolution for the WGS84 model.

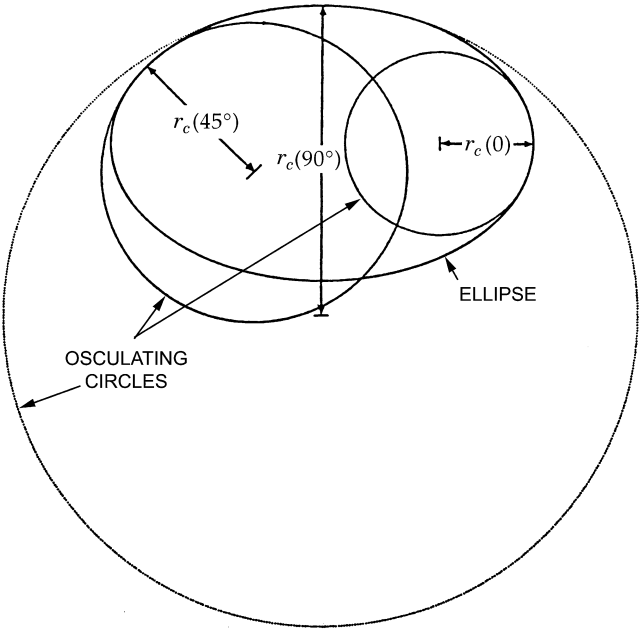


Fig. 3.7 Ellipse and osculating circles.

Radius of curvature can depend on the direction of travel, and for an ellipsoidal model, there is one radius of curvature for north–south motion and another radius of curvature for east–west motion.

Meridional Radius of Curvature The radius of curvature for north–south motion is called the "meridional" radius of curvature, because north–south travel is along a meridian (i.e., line of constant longitude). For an ellipsoid of revolution (the WGS84 model), all meridians have the same shape, which is that of the ellipse that was rotated to produce the ellipsoidal surface model. The tangent circle with the same radius of curvature as the ellipse is called the *osculating circle* (*osculating* means "kissing"). As illustrated in Fig. 3.7 for an oblate earth model, the radius of the meridional osculating circle is smallest where the geocentric radius is largest (at the equator), and the radius of the osculating circle is largest where the geocentric radius is smallest (at the poles). The osculating circle lies inside or on the ellipsoid at the equator and outside or on the ellipsoid at the poles and passes through the ellipsoid surface for latitudes in between.

The formula for meridional radius of curvature as a function of geodetic latitude (ϕ_{geodetic}) is

$$r_M = \frac{b^2}{a[1 - e^2 \sin^2(\phi_{\text{geodetic}})]^{3/2}} \tag{3.10}$$

$$= \frac{a(1-e^2)}{\left[1-e^2\sin^2(\phi_{\text{geodetic}})\right]^{3/2}}, \tag{3.11}$$

where a is the semimajor axis of the ellipse, b is the semiminor axis, and $e^2 = (a^2 - b^2)/a^2$ is the eccentricity squared.

Geodetic Latitude Rate The rate of change of geodetic latitude as a function of north velocity v_N is then

$$\frac{d\phi_{\text{geodetic}}}{dt} = \frac{v_N}{r_M + h}, \tag{3.12}$$

and geodetic latitude can be maintained as the integral

$$\phi_{\text{geodetic}}(t_{\text{now}}) = \phi_{\text{geodetic}}(t_{\text{start}}) + \int_{t_{\text{start}}}^{t_{\text{now}}} \frac{v_{N(t)dt}}{a(1-e^2)/\{1-e^2\sin^2[\phi_{\text{geodetic}}(t)]\}^{3/2} + h(t)]}, \tag{3.13}$$

where $h(t)$ is height above (+) or below (–) the ellipsoid surface and $\phi_{\text{geodetic}}(t)$ will be in radians if $v_N(t)$ is in meters per second and $r_M(t)$ and $h(t)$ are in meters.

Transverse Radius of Curvature The radius of curvature of the reference ellipsoid surface in the east–west direction (i.e., orthogonal to the direction in which the meridional radius of curvature is measured) is called the *transverse radius of curvature*. It is the radius of the osculating circle in the local east–up plane, as illustrated in Fig. 3.8, where the arrows at the point of tangency of the

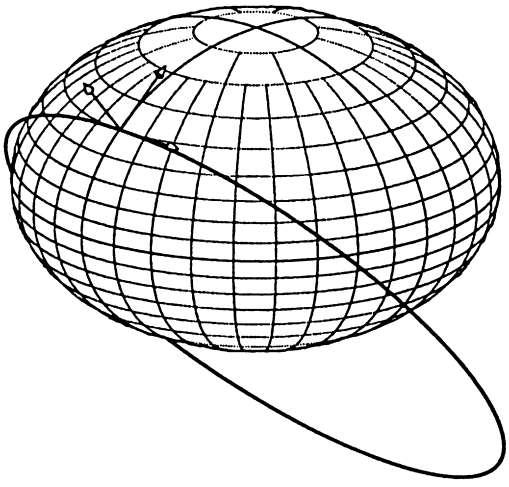


Fig. 3.8 Transverse osculating circle.

transverse osculating circle are in the local ENU coordinate directions. As this figure illustrates, on an oblate earth, the plane of a transverse osculating circle does not pass through the center of the earth except when the point of osculation is at the equator. (All osculating circles at the poles are in meridional planes.) Also, unlike meridional osculating circles, transverse osculating circles generally lie outside the ellipsoidal surface, except at the point of tangency and at the equator, where the transverse osculating circle *is* the equator.

The formula for the transverse radius of curvature on an ellipsoid of revolution is

$$r_T = \frac{a}{\sqrt{1 - e^2 \sin^2(\phi_{\text{geodetic}})}}, \tag{3.14}$$

where a is the semimajor axis of the generating ellipse and e is its eccentricity.

LONGITUDE RATE The rate of change of longitude as a function of east velocity is then

$$\frac{d\theta}{dt} = \frac{v_E}{\cos(\phi_{\text{geodetic}})(r_T + h)}, \tag{3.15}$$

and longitude can be maintained by the integral

$$\theta(t_{\text{now}}) = \theta(t_{\text{start}}) + \int_{t_{\text{start}}}^{t_{\text{now}}} \frac{v_E(t)\,dt}{\cos[\phi_{\text{geodetic}}(t)]\left(a / \sqrt{1 - e^2 \sin^2\left(\phi_{\text{geodetic}}(t)\right)} + h(t)\right)}, \tag{3.16}$$

where $h(t)$ is height above (+) or below (–) the ellipsoid surface and θ will be in radians if $v_E(t)$ is in meters per second and $r_T(t)$ and $h(t)$ are in meters. Note that this formula has a singularity at the poles, where $\cos(\phi_{\text{geodetic}}) = 0$, a consequence of using latitude and longitude as location variables.

WGS84 Reference Surface Curvatures The apparent variations in meridional radius of curvature in Fig. 3.7 are rather large because the ellipse used in generating Fig. 3.7 has an eccentricity of about 0.75. The WGS84 ellipse has an eccentricity of about 0.08, with geocentric, meridional, and transverse radius of curvature as plotted in Fig. 3.9 versus geodetic latitude. For the WGS84 model,

- mean geocentric radius is about 6371 km, from which it varies by –14.3 km (–0.22%) to +7.1 km (+0.11%);
- mean meridional radius of curvature is about 6357 km, from which it varies by –21.3 km (–0.33%) to 42.8 km (+0.67%); and
- mean transverse radius of curvature is about 6385 km, from which it varies by –7.1 km (–0.11%) to +14.3 km (+0.22%).

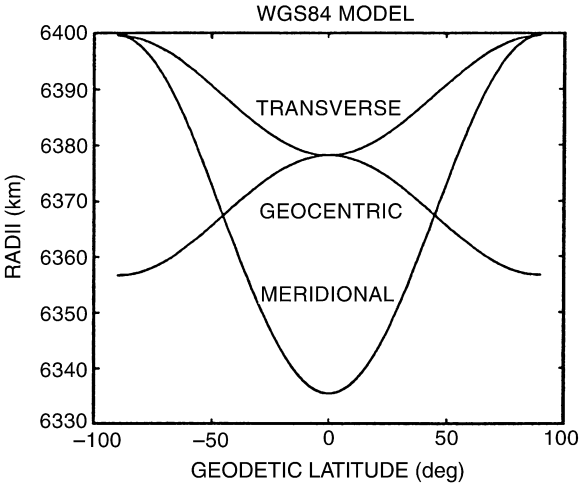


Fig. 3.9 Radii of WGS84 reference ellipsoid.

Because these vary by several parts per thousand, one must take the radius of curvature into account when integrating horizontal velocity increments to obtain the longitude and the latitude.

3.6 HARDWARE IMPLEMENTATIONS

INSs generally fall into two categories depending on the hardware configuration:

1. **Gimbaled** or **floated** systems, in which the ISA is isolated from rotations of the host vehicle, as illustrated in Fig. 3.10(*a*–*c*). This shows three alternative structures that have been tried at different times:
   (a) Gimbals, also called a Cardan[4] suspension. This is the most popular implementation using hardware to solve the attitude problem.
   (b) Ball joint, which Fritz Mueller called "inverted gimbals" [7]. It has not become popular, perhaps because of the difficulties of applying controlled torques about the spherical bearing to stabilize the ISA. This configuration is not discussed here.
   (c) A floated sphere, a configuration also called "FLIMBAL," an acronym for floated inertial measurement ball.[5] Despite the difficul-

[4]Named after the Italian physician, inventor, and polymath Girolamo Cardano (1501–1576), who also invented what Americans call a "universal joint" and Europeans call a "Cardan shaft."
[5]A name used at the MIT Instrumentation Laboratory around 1957, when work on its development started.

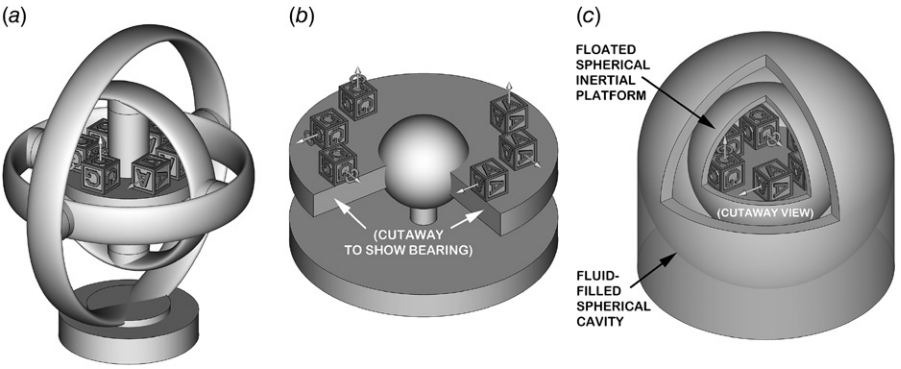


Fig. 3.10 Gimbaled IMU alternatives. (*a*) Gimbal (Cardan suspension). (*b*) Ball joint ("inverted gimbals"). (*c*) Floated sphere ("FLIMBAL").

ties of transferring power, signals, heat, torque, and relative attitude between the housing and the inner spherical ISA, it is probably the most accurate (and expensive) implementation for high-g rocket booster applications.

In all cases, the rotation-isolated ISA is also called an inertial platform, stable platform, or stable element. The IMU includes the ISA, the gimbal/float structure, and all associated electronics (e.g., gimbal wiring, rotary slip rings, gimbal bearing angle encoders, signal conditioning, gimbal bearing torque motors, and thermal control).

2. **Strapdown** systems, essentially as illustrated in Fig. 3.1. In this case, the ISA is not isolated from rotations but is "quasi-rigidly" mounted to the frame structure of the host vehicle.

We use the term "quasi-rigid" for IMU mountings that can provide some isolation of the IMU from shock and vibration transmitted through the host vehicle frame. These vibrational accelerations do not significantly alter the navigation solution, but they can damage the ISA and its sensors. Strapdown, gimbaled, and floated systems may require **shock and vibration isolators** to dampen the vibrational torques and forces transmitted to the inertial sensors. These isolators are commonly made from "lossy" elastomers that provide some amount of damping, as well.

3.6.1 Gimbaled Implementations

The use of gimbals for isolation from rotation has been documented as far back as the third century BCE. The term generally applies to the entire structure, usually consisting of a set of two or three (or four) nested rings with orthogonal pivots (also called "gimbal bearings"). As illustrated in Fig. 3.11(*a*), three sets of gimbal bearings are sufficient for complete rotational isolation in applications with limited attitude mobility (e.g., surface ships), but applications

in fully maneuverable host vehicles require an additional gimbal bearing to avoid the condition shown in Fig. 3.11(*b*), known as "gimbal lock," in which the gimbal configuration no longer provides isolation from outside rotations about all three axes.

For inertial navigation, gyroscopes inside the gimbals detect any incipient rotation of that frame due to torques from any source (e.g., bearing friction or mass imbalance) and apply feedback to torquing motors in the gimbal bearings to keep the rotation rates inside the gimbals at zero. For navigation with respect to the rotating earth, the gimbals can also be servoed to maintain the sensor axes fixed in locally level coordinates.

Design of gimbal torquing servos is complicated by the motions of the gimbals during operation, which changes how the torquing to correct for sensed rotation must be applied to the different gimbal bearings. This requires a bearing angle sensor for each gimbal axis.

The gimbal arrangement shown in Fig. 3.11(*a*), with the outer gimbal axis aligned to the roll (longitudinal) axis of the host vehicle and the inner gimbal axis maintained in the vertical direction, is a popular one. If the ISA is kept aligned with locally level ENU directions, the gimbal bearing angles will equal the heading (yaw), pitch, and roll Euler angles defining the host vehicle attitude relative to north, east, and down directions. These are the same Euler angles used to drive **attitude and heading reference systems** (AHRSs) (e.g., compass card and artificial horizon displays) in aircraft cockpits.

Advantages The principal advantage of both gimbaled and floated systems is the isolation of the inertial sensors from high angular rates, which eliminates many rate-dependent sensor errors (including gyro scale factor sensitivity) and generally allows for higher accuracy sensors. Also, gimbaled systems can be self-calibrated by orienting the ISA with respect to gravity (for calibrating the accelerometers) and with respect to the earth rotation axis (for

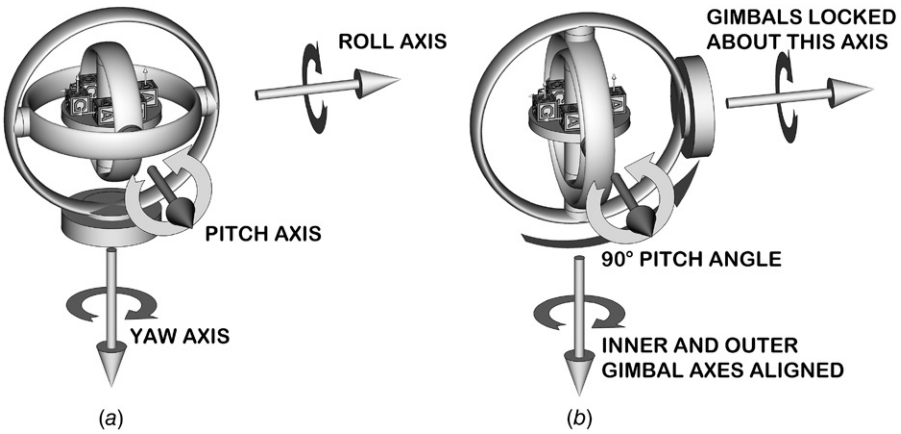


Fig. 3.11 Three-axis gimbal lock. (*a*) Straight and level. (*b*) Pitched up 90°.

calibrating the gyros), and by using external optical autocollimators with mirrors on the ISA to independently measure its orientation with respect to its environment.

The most demanding INS applications for "cruise" applications (i.e., at ≈1 g) are probably for nuclear missile-carrying submarines, which must navigate submerged for months. The gimbaled Electrically Supported Gyro Navigation (ESGN, DoD designation AN/WSN-3 [2]) system developed in the 1970s for USN Trident-class submarines was probably the most accurate INS of that era [6].

Disadvantages The principal drawbacks of gimbals are cost, weight, volume, and gimbal flexure in high-*g* environments. In traditional designs, electrical pathways are required through the gimbal structure to provide power to the IMU and to carry power and the encoder, torquer, and sensor signals. These require slip rings (which can introduce noise) or cable wraps at the gimbal bearings. More recent designs, however, have used wireless signal transmission. The gimbals can alter air circulation used to maintain uniform temperatures within the IMU, and they can hamper access to the sensors during test and operation (e.g., optical measurements using mirrors on the ISA to check its attitude).

3.6.2 Floated Implementation

Gimbals and gimbal lock can be eliminated by floating the ISA in a liquid and operating it like a robotic submersible, using liquid thrusters to maintain its orientation and to keep itself centered within the flotation cavity—as illustrated on the right in Fig. 3.10. The floated assembly must also be neutrally buoyant and balanced to eliminate acceleration-dependent disturbances.

Advantages Floated systems have the advantage over gimbaled systems that there are no gimbal structures to flex or vibrate under dynamic loading, and no gimbals interfering with heat transfer from the ISA. Floated systems have the same advantages as gimbaled systems over strapdown systems: isolation of the inertial sensors from high angular rates, which eliminates many rate-dependent error effects and generally allows for higher accuracy sensors. They also have the ability to orient the sphere for self-calibration, although it is more difficult to verify the orientation of the ISA from the outside. The floated Advanced Inertial Reference Sphere (AIRS) designed at the Charles Stark Draper Laboratory for MX/Peacekeeper and Minuteman III missiles is probably the most accurate (and most expensive) high-g INS ever developed [6].

Disadvantages A major disadvantage of floated systems is the difficulty of accessing the ISA for diagnostic testing, maintenance, or repair. The flotation system must be disassembled and the fluid drained for access, and then reassembled for operation. Floated systems also require some means for determining the attitude of the floated assembly relative to the host vehicle and providing power to the floated assembly, and passing commands and sensor

signals through the fluid, and for maintaining precisely controlled temperatures within the floated IMU.

Also, for applications in which the host vehicle attitude must be controlled, gimbaled or floated systems provide only vehicle attitude information, whereas strapdown systems provide attitude rates for vehicle attitude control loops.

3.6.3 Carouseling and Indexing

3.6.3.1 Alpha Wander and Carouseling *Alpha Wander* Near the poles, there is a problem with locally level gimbal orientations in which the level axes are constrained to point north and east. At the poles, there is no north or east direction; all directions are either south (at the North Pole) or north (at the South Pole). Also, near the poles the slewing rates required to keep the north/east orientations can be unreasonably high. The solution is to allow the locally level axes to wander, with the angle α (alpha) designating the angle between one of the level axes and north (except near the poles, where it is referenced to earth-fixed polar stereographic coordinates). This is called an "alpha wander" implementation.

Carouseling A *carousel* is an amusement ride using continuous rotation of a circular platform about a vertical axis. The term "carouseling" has been applied to an implementation for gimbaled or floated systems in which the ISA revolves slowly around the local vertical axis—at rates in the order of a revolution per minute. The three-gimbal configuration shown on the left in Fig. 3.11(*a*) can implement carouseling using only the inner (vertical) gimbal axis. Carouseling significantly reduces long-term navigation errors due to certain types of sensor errors (uncompensated biases of nominally level accelerometers and gyroscopes, in particular). This effect was discovered and exploited at the AC[6] Spark Plug (later Delco Electronics) Division of General Motors in the early 1960s. Delco Carousel systems became phenomenally successful and popular for years.

3.6.3.2 Indexing Alternative implementations called **indexing** or **gimbal flipping** use discrete rotations (usually by multiples of 90°) to the same effect.

Indexing can also be used at the sensor level. The electrostatic gyroscopes in the U.S. Navy's ESGN (DoD designation AN/WSN-3 [2]) use independent gimbals to keep the spin axis of each rotor in the same direction relative to its suspension cavity but index those gimbals[7] to provide periodic cavity rotations. The resulting system accuracy is classified, but good enough that nothing was able to surpass it for decades.

[6]The "AC" in the name is the initials of Albert Champion (1878–1927), a former world-class cyclist who founded Champion Spark Plug Company in Boston in the early 1900s. When he lost control of that company in 1908, Albert founded the Champion Ignition Company in Flint, Michigan. It was renamed the AC Spark Plug Company in 1909, when it was purchased by General Motors.
[7]Using a pattern designed by Kenneth P. Gow (1917–2001), and called the "Gow flip."

3.6.4 Strapdown Systems

Strapdown systems use an IMU that is not isolated from rotations of its host vehicle—except possibly by shock and vibration isolators. The gimbals are effectively replaced by software that uses the gyroscope outputs to calculate the equivalent accelerometer outputs in an attitude-stabilized coordinate frame, and integrates them to provide updates of velocity and position. This requires more computation (which is cheap) than the gimbaled implementation, but it eliminates the gimbal system (which may not be cheap). It also exposes the accelerometers and gyroscopes to relatively high rotation rates, which can cause attitude rate-dependent sensor errors.

Advantages The principal advantage of strapdown systems over gimbaled or floated systems is cost. The cost or replicating software is vanishingly small compared to the cost of replicating a gimbal system for each IMU. For applications requiring attitude control of the host vehicle, strapdown gyroscopes generally provide more accurate rotation rate data than the attitude readouts of gimbaled or floated systems.

Disadvantages Strapdown sensors must operate at much higher rotation rates, which limits the possible design choices. Pendulous integrating gyroscopic accelerometers, for example, are very sensitive to rotation. The dynamic ranges of the inputs to strapdown gyroscopes may be orders of magnitude greater than those for gyroscopes in gimbaled systems. To achieve comparable navigation performance, this generally requires orders of magnitude better scale factor stability for the strapdown gyroscopes. Strapdown systems generally require much shorter integration intervals—especially for integrating gyroscope outputs, which increases computer costs relative to gimbaled systems. Another disadvantage for strapdown is the cost of gyroscope calibration and testing, which requires a precision rate table. Precision rate table testing is not required for whole-angle gyroscopes—including electrostatic gyroscopes—or for any gyroscopes used in gimbaled systems.

3.6.5 Strapdown Carouseling and Indexing

For host vehicles that are nominally upright during operation (e.g., ships), a strapdown system can be rotated about the host vehicle yaw axis. So long as the vehicle yaw axis remains close to the local vehicle, slow rotation (carouseling) or indexing about this axis can significantly reduce the effects of uncompensated biases of the nominally level accelerometers and gyroscopes. The rotation is normally oscillatory, with reversal of direction after a full rotation, so that the connectors can be wrapped to avoid using slip rings (an option not generally available for gimbaled systems).

Carouseling or indexing of strapdown systems requires the addition of a rotation bearing and associated motor drive, wiring and control electronics. However, the improvement in navigational performance may justify the additional cost. The U.S. Air Force N–73 inertial navigator was an early strapdown system using carouseling, and its navigation accuracy was better than one nautical mile per hour CEP rate.

3.7 SOFTWARE IMPLEMENTATIONS

3.7.1 Example in One Dimension

Inertial navigation would be much simpler if we all lived in a one-dimensional "line land." For one thing, there would be no rotation and no need for gyroscopes or gimbals. In that case, an INS would need only one accelerometer and navigation computer (all one-dimensional line segments, of course), and its implementation would be about as illustrated in Fig. 3.12 (in two dimensions), where the dependent variable x denotes position on the line in one dimension and the independent variable t is time.

This implementation for one dimension still has some features in common with implementations for three dimensions:

1. Accelerometers cannot measure gravitational acceleration.
2. Accelerometers have *scale factors*, which are the ratios of input acceleration units to output signal magnitude units (e.g., meters per second squared per volt). The signal must be rescaled in the navigation computer by multiplying by this scale factor.
3. Accelerometers have *output errors*, including
   (a) unknown constant *offsets*, also called *biases*;
   (b) unknown constant *scale factor errors*;
   (c) unknown *nonconstant variations* in bias and scale factor; and
   (d) unknown zero-mean additive *noise* on the sensor outputs, including quantization noise and electronic noise. The noise itself is not predictable, but its statistical properties may be used in Kalman filtering to estimate drifting scale factor and biases.

   In one dimension, there is no such thing as input axis misalignment.
4. Gravitational accelerations must be modeled and calculated in the navigational computer, then added to the sensed acceleration (after

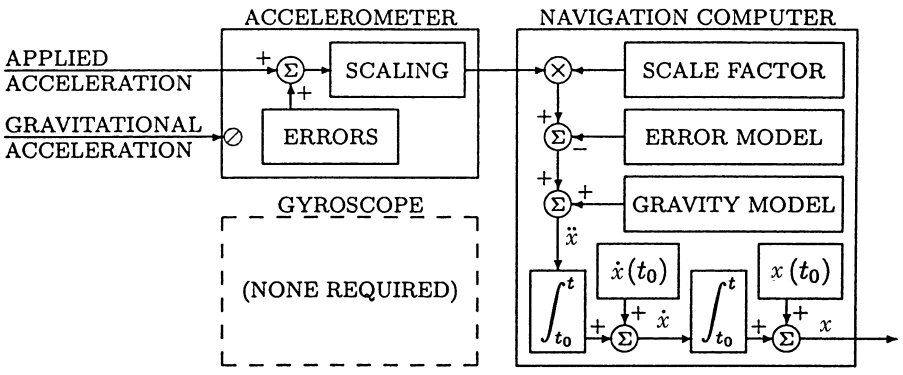


Fig. 3.12 INS functional implementation for a one-dimensional world.

error and scale compensation) to obtain the net acceleration $\ddot{x}$ of the INS.

5. The navigation computer must integrate acceleration to obtain velocity. This is a definite integral and it requires an initial value, $\dot{x}(t_0)$; that is, the INS implementation in the navigation computer must start with a known initial velocity.
6. The navigation computer must also integrate velocity ($\dot{x}$) to obtain position (x). This is also a definite integral and it also requires an initial value, $x(t_0)$. The INS implementation in the navigation computer must start with a known initial location too.

Inertial navigation in three dimensions requires more sensors and more signal processing than in one dimension, and it also introduces more possibilities for implementation (e.g., gimbaled or strapdown).

3.7.2 Initialization in Nine Dimensions

In the three-dimensional world of inertial navigation, the navigation solution requires initial values for

1. position (three-dimensional),
2. velocity (also three-dimensional), and
3. attitude (also three-dimensional).

3.7.2.1 Navigation Initialization INS initialization is the process of determining initial values for system position, velocity, and attitude in navigation coordinates. INS position initialization ordinarily relies on external sources such as GNSS, local wireless service, or manual entry. INS velocity initialization can be accomplished by starting when it is zero (i.e., the host vehicle is not moving) or (for vehicles carried in or on other vehicles) by reference to the carrier velocity. (See alignment method 3 below.) INS attitude initialization is called **alignment**.

3.7.2.2 INS Alignment Methods INS alignment is the process of determining the ISA orientation relative to navigation coordinates.

There are four basic alignment methods:

1. **Optical alignment**, using either of the following:
   (a) Optical line-of-sight reference to a ground-based direction (e.g., using a ground-based theodolite and a mirror on the platform). Some space boosters have used this type of optical alignment, which is much faster and more accurate than gyrocompass alignment. Because it requires a stable platform for mounting the mirror, it is only applicable to gimbaled systems.

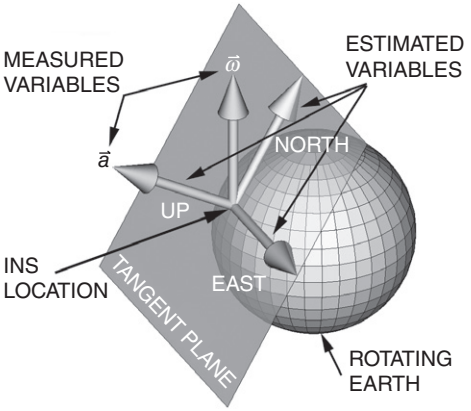


Fig. 3.13 Gyrocompassing determines sensor orientations with respect to east, north, and up.

(b) An onboard star tracker, used primarily for alignment of gimbaled or strapdown systems in space or near space (e.g., above the clouds).

2. **Gyrocompass alignment** of stationary vehicles, using the sensed direction of acceleration to determine the local vertical and the sensed direction of rotation to determine north, as illustrated in Fig. 3.13. Latitude can be determined by the angle between the earth rotation vector and the horizontal, but longitude must be determined by other means and entered manually or electronically. This method is inexpensive but the most time-consuming (several minutes, typically).
3. **Transfer alignment** in a moving host vehicle, using velocity matching with an aligned and operating INS. This method is generally faster than gyrocompass alignment, but it requires another INS on the host vehicle and it may require special maneuvering of the host vehicle to attain observability of the alignment variables. It is commonly used for in-air INS alignment for missiles launched from aircraft and for on-deck INS alignment for aircraft launched from carriers. Alignment of carrier-launched aircraft may also use the direction of the velocity impulse imparted by the steam catapult.
4. **GNSS-aided alignment**, using position matching with GNSS to estimate the alignment variables. It is an integral part of integrated GNSS/INS implementations. It does not require the host vehicle to remain stationary during alignment, but there will be some period of time after turn-on (a few minutes, typically) before system navigation errors settle to acceptable levels.

3.7.2.3 Gyrocompass Alignment Gyrocompass alignment is the only one of those listed above requiring no external aiding. Gyrocompass alignment is

not necessary for integrated GNSS/INS, although many INSs may already be configured for it.

Accuracy A rough rule of thumb for gyrocompass alignment accuracy is

$$\sigma^2_{\text{gyrocompass}} > \sigma^2_{\text{acc}} + \frac{\sigma^2_{\text{gyro}}}{15^2 \cos^2(\phi_{\text{geodetic}})}, \tag{3.17}$$

where

$\sigma_{\text{gyrocompass}}$ is the minimum achievable RMS alignment error in radians,
σ_{acc} is the RMS accelerometer accuracy in g's,
σ_{gyro} is the RMS gyroscope accuracy in degrees per hour,
15 deg/h is the rotation rate of the earth, and
ϕ_{geodetic} is the latitude at which gyrocompassing is performed.

Alignment accuracy is also a function of the time allotted for it, and the time required to achieve a specified accuracy is generally a function of sensor error magnitudes (including noise) and the degree to which the vehicle remains stationary.

Gimbaled Implementation Gyrocompass alignment for gimbaled systems is a process for aligning the inertial platform axes with the navigation coordinates using only the sensor outputs, while the host vehicle is essentially stationary. For systems using ENU navigation coordinates, for example, the platform can be tilted until two of its accelerometer inputs are zero, at which time both input axes will be horizontal. In this locally leveled orientation, the sensed rotation axis will be in the north–up plane, and the platform can be slewed about the vertical axis to null the input of one of its horizontal gyroscopes, at which time that gyroscope input axis will point east–west. That is the basic concept used for gyrocompass alignment, but practical implementation requires filtering[8] to reduce the effects of sensor noise and unpredictable zero-mean vehicle disturbances due to loading activities and/or wind gusts.

Strapdown Implementation Gyrocompass alignment for strapdown systems is a process for "virtual alignment" by determining the sensor cluster attitude with respect to navigation coordinates using only the sensor outputs while the system is essentially stationary.

Error-Free Implementation If the sensor cluster could be firmly affixed to the earth and there were no sensor errors, then the sensed acceleration vector

[8]The vehicle dynamic model used for gyrocompass alignment filtering can be "tuned" to include the major resonance modes of the vehicle suspension.

$\mathbf{a}_{\text{output}}$ in sensor coordinates would be in the direction of the local vertical, the sensed rotation vector $\boldsymbol{\omega}_{\text{output}}$ would be in the direction of the earth rotation axis, and the unit column vectors

$$1_U = \frac{\mathbf{a}_{\text{output}}}{|\mathbf{a}_{\text{output}}|}, \tag{3.18}$$

$$1_N = \frac{\boldsymbol{w}_{\text{output}} - (1_U^T \boldsymbol{w}_{\text{output}})1_U}{|\boldsymbol{w}_{\text{output}} - (1_U^T \boldsymbol{w}_{\text{output}})1_U|}, \tag{3.19}$$

$$1_E = 1_N \otimes 1_U \tag{3.20}$$

would define the initial value of the coordinate transformation matrix from sensor-fixed coordinates to ENU coordinates:

$$\mathbf{C}_{\text{ENU}}^{\text{sensor}} = [1_E | 1_N | 1_U]^T. \tag{3.21}$$

Practical Implementation In practice, the sensor cluster is usually mounted in a vehicle that is not moving over the surface of the earth but may be buffeted by wind gusts or disturbed during fueling and loading operations. Gyrocompassing then requires some amount of filtering (Kalman filtering, as a rule) to reduce the effects of vehicle buffeting and sensor noise. The gyrocompass filtering period is typically on the order of several minutes for a medium-accuracy INS but may continue for hours, days, or continuously for high-accuracy systems.

3.7.3 Gimbal Attitude Implementations

The primary function of gimbals is to isolate the ISA from vehicle rotations, but they are also used for other INS functions.

3.7.3.1 Accelerometer Recalibration Navigation accuracy is very sensitive to accelerometer biases, which can shift due to thermal transients in turn-on/turn-off cycles, and can also drift randomly over time. Fortunately, the gimbals can be used to calibrate accelerometer biases in a stationary 1-g environment. In fact, both bias and scale factor can be determined by using the gimbals to point the accelerometer input axis straight up and straight down, and recording the respective accelerometer outputs a_{up} and a_{down}. Then the bias $a_{\text{bias}} = (a_{\text{up}} + a_{\text{down}})/2$ and scale factor $s = (a_{\text{up}} - a_{\text{down}})/2g_{\text{local}}$, where g_{local} is the local gravitational acceleration.

3.7.3.2 Vehicle Attitude Determination The gimbal angles determine the vehicle attitude with respect to the ISA, which has a controlled orientation with respect to navigation coordinates. Each gimbal angle encoder output determines the relative rotation of the structure outside the gimbal axis relative to the structure inside the gimbal axis; the effect of each rotation can be

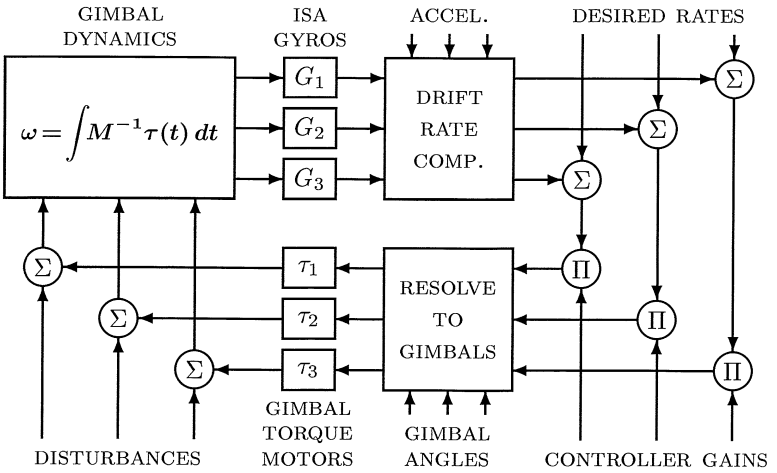


Fig. 3.14 Simplified control flow diagram for three gimbals.

represented by a 3×3 rotation matrix, and the coordinate transformation matrix representing the attitude of the vehicle with respect to the ISA will be the ordered product of these matrices.

For example, in the gimbal structure shown in Fig. 3.11(*a*), each gimbal angle represents an Euler angle for vehicle rotations about the vehicle roll, pitch, and yaw axes.

3.7.3.3 ISA Attitude Control Gimbals control ISA orientation. This is a 3-degree-of-freedom problem, and the solution is unique for three gimbals; that is, there are three attitude control loops with (at least) three sensors (the gyroscopes) and three torquers. Each control loop can use a proportional, integral, and differential (PID) controller, with the commanded torque distributed to the three torquers according to the direction of the torquer/gimbal axis with respect to the gyro input axis, somewhat as illustrated in Fig. 3.14, where

DISTURBANCES includes the sum of all torque disturbances on the individual gimbals and the ISA, including those due to ISA mass unbalance and acceleration, rotations of the host vehicle, air currents, and torque motor errors.

GIMBAL DYNAMICS is actually quite a bit more complicated than the rigid-body torque equation

$$\boldsymbol{\tau} = \mathbf{M}_{\text{inertia}}\dot{\boldsymbol{\omega}},$$

which is the torque analog of $\mathbf{F} = m\mathbf{a}$, where $\mathbf{M}_{\text{inertia}}$ is the moment of inertia matrix. The IMU is not a rigid body, and the gimbal torque motors apply torques *between* the gimbal elements (i.e., ISA, gimbal rings, and host vehicle).

DESIRED RATES refer to the rates required to keep the ISA aligned to a moving coordinate frame (e.g., locally level).

RESOLVE TO GIMBALS is where the required torques are apportioned among the individual torquer motors on the gimbal axes. The actual control loop is more complicated than that shown in the figure, but it does illustrate in general terms how the sensors and actuators are used.

For systems using four gimbals to avoid gimbal lock, the added gimbal adds another degree of freedom to be controlled. In this case, the control law usually adds a fourth constraint (e.g., maximize the minimum angle between gimbal axes) to avoid gimbal lock.

3.7.4 Gimbaled Navigation Implementation

The signal flowchart in Fig. 3.15 shows the essential navigation signal processing functions for a gimbaled INS with inertial sensor axes aligned to locally level coordinates, where

$\boldsymbol{f}_{\text{specific}}$ is the **specific force** (i.e., the sensible acceleration, exclusive of gravitational acceleration) applied to the host vehicle.

Ω_{inertial} is the instantaneous inertial rotation rate vector of the host vehicle.

A denotes a specific force sensor (accelerometer).

θ_j denotes the ensemble of gimbal angle encoders, one for each gimbal angle. There are several possible formats for the gimbal angles, including digitized angles, three-wire synchros signals, or `sin/cos` pairs.

G denotes an inertial rotation rate sensor (gyroscope).

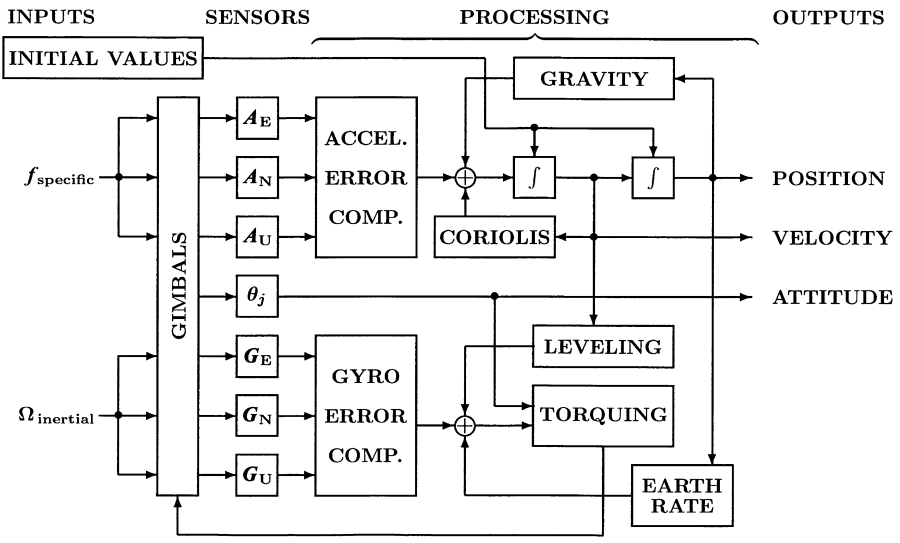


Fig. 3.15 Essential navigation signal processing for gimbaled INS.

POSITION is the estimated position of the host vehicle in navigation coordinates (e.g., longitude, latitude, and altitude relative to sea level).

VELOCITY is the estimated velocity of the host vehicle in navigation coordinates (e.g., east, north, and vertical).

ATTITUDE is the estimated attitude of the host vehicle relative to locally level coordinates. For some three-gimbal systems, the gimbal angles are the Euler angles representing vehicle heading (with respect to north), pitch, and roll. Output attitude may also be used to drive cockpit displays such as compass cards or artificial horizon indicators.

ACCELEROMETER ERROR COMPENSATION and GYROSCOPE ERROR COMPENSATION denote the calibrated corrections for sensor errors. These generally include corrections for scale factor variations, output biases and input axis misalignments for both types of sensors, and acceleration-dependent errors for gyroscopes.

GRAVITY denotes the gravity model used to compute the acceleration due to gravity as a function of position.

CORIOLIS denotes the acceleration correction for coriolis effect in rotating coordinates.

LEVELING denotes the rotation rate correction to maintain locally level coordinates while moving over the surface of the earth.

EARTHRATE denotes the model used to calculate the earth rotation rate in locally level INS coordinates.

TORQUING denotes the servo loop gain computations used in stabilizing the INS in locally level coordinates.

Not shown in the figure is the input altitude reference (e.g., barometic altimeter or GPS) required for vertical channel (altitude) stabilization.

3.7.5 Strapdown Attitude Implementations

3.7.5.1 Strapdown Attitude Problems Early on, strapdown systems technology had an "attitude problem," which was the problem of representing attitude rate in a format amenable to accurate computer integration. The eventual solution was to represent attitude in different mathematical formats as it is processed from raw gyro outputs to the matrices used for transforming sensed acceleration to inertial coordinates for integration.

Figure 3.16 illustrates the resulting major gyro signal processing operations and the formats of the data used for representing attitude information. The processing starts with gyro outputs and ends with a coordinate transformation matrix from sensor coordinates to the coordinates used for integrating the sensed accelerations.

3.7.5.2 Coning Motion This type of motion is a problem for attitude integration when the frequency of motion is near or above the sampling frequency.

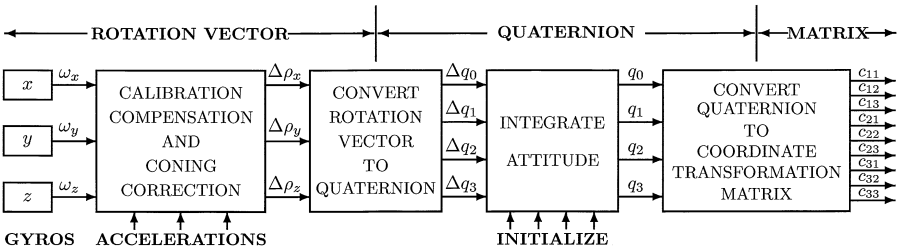


Fig. 3.16 Strapdown attitude representations.

It is usually a consequence of host vehicle frame vibration modes where the INS is mounted, and INS shock and vibration isolation is often designed to eliminate or substantially reduce this type of rotational vibration.

Coning motion is an example of an attitude trajectory (i.e., attitude as a function of time) for which the integral of attitude rates does *not* equal the attitude change. An example trajectory would be

$$\boldsymbol{\rho}(t) = \theta_{\text{cone}} \begin{bmatrix} \cos(\Omega_{\text{coning}} t) \\ \sin(\Omega_{\text{coning}} t) \\ 0 \end{bmatrix} \tag{3.22}$$

$$\dot{\boldsymbol{\rho}}(t) = \theta_{\text{cone}} \Omega_{\text{coning}} \begin{bmatrix} -\sin(\Omega_{\text{coning}} t) \\ \cos(\Omega_{\text{coning}} t) \\ 0 \end{bmatrix}, \tag{3.23}$$

where

θ_{cone} is called the *cone angle* of the motion,

Ω_{coning} is the *coning frequency* of the motion, as illustrated in Fig. 3.17.

The coordinate transformation matrix from body coordinates to inertial coordinates (Eq. B.112 of Appendix B) will be

$$\begin{aligned} \mathbf{C}_{\text{inertial}}^{\text{body}}(\boldsymbol{\rho}) &= \cos\theta \mathbf{I} \\ &+ (1-\cos\theta) \begin{bmatrix} \cos(\Omega_{\text{coning}} t)^2 & \sin(\Omega_{\text{coning}} t)\cos(\Omega_{\text{coning}} t) & 0 \\ \sin(\Omega_{\text{coning}} t)\cos(\Omega_{\text{coning}} t) & \sin(\Omega_{\text{coning}} t)^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ &+ \sin\theta \begin{bmatrix} 0 & 0 & \sin(\Omega_{\text{coning}} t) \\ 0 & 0 & -\cos(\Omega_{\text{coning}} t) \\ -\sin(\Omega_{\text{coning}} t) & \cos(\Omega_{\text{coning}} t) & 0 \end{bmatrix}, \end{aligned} \tag{3.24}$$

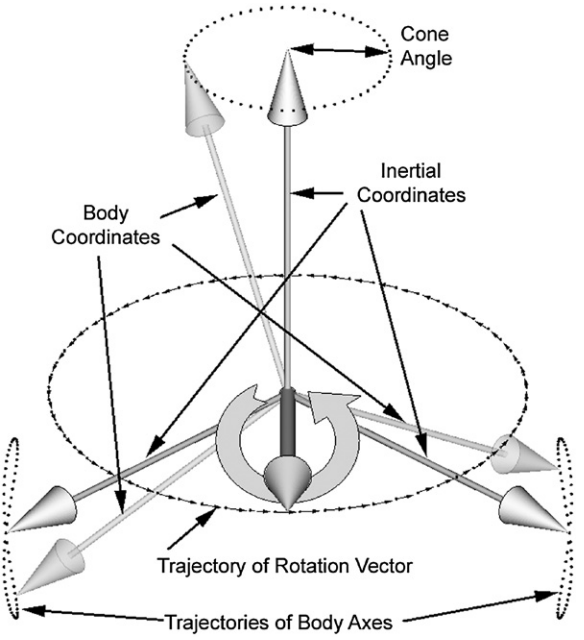


Fig. 3.17 Coning motion.

and the measured inertial rotation rates in body coordinates will be

$$\boldsymbol{\omega}_{\text{body}} = \mathbf{C}_{\text{body}}^{\text{inertial}} \dot{\boldsymbol{\rho}}_{\text{inertial}} \tag{3.25}$$

$$= \theta_{\text{cone}} \Omega_{\text{coning}} \left[\mathbf{C}_{\text{inertial}}^{\text{body}} \right]^T \begin{bmatrix} -\sin(\Omega_{\text{coning}} t) \\ \cos(\Omega_{\text{coning}} t) \\ 0 \end{bmatrix} \tag{3.26}$$

$$= \begin{bmatrix} -\theta_{\text{cone}} \Omega_{\text{coning}} \sin(\Omega_{\text{coning}} t) \cos(\theta_{\text{cone}}) \\ \theta_{\text{cone}} \Omega_{\text{coning}} \cos(\Omega_{\text{coning}} t) \cos(\theta_{\text{cone}}) \\ -\sin(\theta_{\text{cone}}) \theta_{\text{cone}} \Omega_{\text{coning}} \end{bmatrix}. \tag{3.27}$$

The integral of $\boldsymbol{\omega}_{\text{body}}$

$$\int_{s=0}^{t} \boldsymbol{\omega}_{\text{body}}(s) ds = \begin{bmatrix} -\theta_{\text{cone}} \cos(\theta_{\text{cone}}) [1 - \cos(\Omega_{\text{coning}} t)] \\ \theta_{\text{cone}} \cos(\theta_{\text{cone}}) \sin(\Omega_{\text{coning}} t) \\ -\sin(\theta_{\text{cone}}) \theta_{\text{cone}} \Omega_{\text{coning}} t \end{bmatrix}, \tag{3.28}$$

which is what a rate integrating gyroscope would measure.

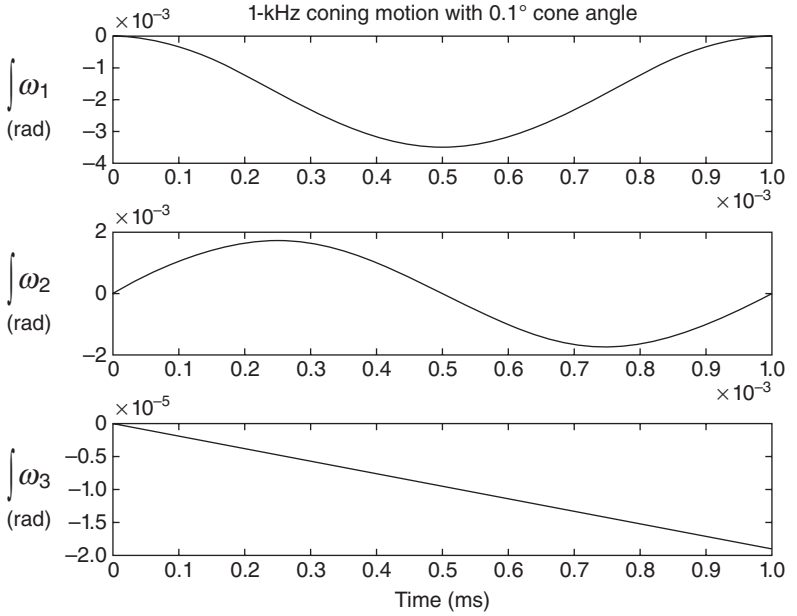


Fig. 3.18 Coning error for 0.1° cone angle, 1-kHz coning rate.

The solutions for $\theta_{\text{cone}} = 0.1°$ and $\Omega_{\text{coning}} = 1\,\text{kHz}$ are plotted over one cycle (1 ms) in Fig. 3.18. The first two components are cyclical, but the third component accumulates linearly over time at about -1.9×10^{-5} rad in 10^{-3} s, which is a bit more than −1 deg/s. *This is why coning error compensation is important.*

3.7.5.3 Rotation Vector Implementation This implementation is primarily used at a faster sampling rate than the nominal sampling rate (i.e., that required for resolving measured accelerations into navigation coordinates). It is used to remove the nonlinear effects of coning and skulling motion that would otherwise corrupt the accumulated angle rates over the nominal intersample period. This implementation is also called a "coning correction."

Bortz Model for Attitude Dynamics This exact model for attitude integration based on measured rotation rates and rotation vectors was developed by John Bortz [1]. It represents ISA attitude with respect to the reference inertial coordinate frame in terms of the rotation vector $\boldsymbol{\rho}$ required to rotate the reference inertial coordinate frame into coincidence with the sensor-fixed coordinate frame, as illustrated in Fig. 3.19.

The Bortz dynamic model for attitude then has the form

$$\dot{\boldsymbol{\rho}} = \boldsymbol{\omega} + \mathbf{f}_{\text{Bortz}}(\boldsymbol{\omega}, \boldsymbol{\rho}), \tag{3.29}$$

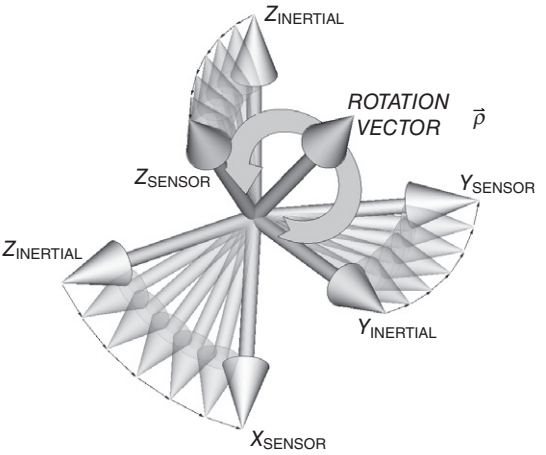


Fig. 3.19 Rotation vector representing coordinate transformation.

where $\boldsymbol{\omega}$ is the vector of measured rotation rates. The Bortz "noncommutative rate vector"

$$\mathbf{f}_{\text{Bortz}}(\boldsymbol{\omega}, \boldsymbol{\rho}) = \frac{1}{2}\boldsymbol{\rho} \otimes \boldsymbol{\omega} + \frac{1}{\|\boldsymbol{\rho}\|^2}\left\{1 - \frac{\|\boldsymbol{\rho}\|\sin(\|\boldsymbol{\rho}\|)}{2[1-\cos(\|\boldsymbol{\rho}\|)]}\right\}\boldsymbol{\rho} \otimes (\boldsymbol{\rho} \otimes \boldsymbol{\omega}) \tag{3.30}$$

$$|\boldsymbol{\rho}| < \frac{\pi}{2}. \tag{3.31}$$

Equation 3.29 represents the rate of change of attitude as a nonlinear differential equation that is linear in the measured instantaneous body rates $\boldsymbol{\omega}$. Therefore, by integrating this equation over the nominal intersample period $[0, \Delta t]$ with initial value $\boldsymbol{\rho}(0) = 0$, an exact solution of the body attitude change over that period can be obtained in terms of the net rotation vector

$$\Delta\boldsymbol{\rho}(\Delta t) = \int_0^{\Delta t} \dot{\boldsymbol{\rho}}(\boldsymbol{\rho}(s), \boldsymbol{\omega}(s))ds, \tag{3.32}$$

which avoids all the noncommutativity errors and satisfies the constraint of Eq. 3.31 so long as the body cannot turn 180° in one sample interval Δt. In practice, the integral is done numerically with the gyro outputs ω_1, ω_2, ω_3 sampled at intervals $\delta t = \Delta t$. The choice of δt is usually made by analyzing the gyro outputs under operating conditions (including vibration isolation), and selecting a sampling freqency $1/\delta t$ well above the Nyquist frequency for the observed attitude rate spectrum. The frequency response of the gyros also enters into this design analysis.

The MATLAB® function `fBortz.m` on the accompanying website calculates $\mathbf{f}_{\text{Bortz}}(\boldsymbol{\omega})$ defined by Eq. 3.30.

3.7.5.4 Quaternion Implementation The quaternion representation of vehicle attitude is the most reliable, and it is used as the "holy point" of attitude representation. Its value is maintained using the incremental rotations $\Delta\boldsymbol{\rho}$ from the rotation vector representation, and the resulting values are used to generate the coordinate transformation matrix for accumulating velocity changes in inertial coordinates.

Quaternions represent three-dimensional attitude on the three-dimensional surface of the four-dimensional sphere, much like two-dimensional directions can be represented on the two-dimensional surface of the three-dimensional sphere.

Converting Incremental Rotations to Incremental Quaternions An incremental rotation vector $\Delta\boldsymbol{\rho}$ from the Bortz coning correction implementation of Eq. 3.32 can be converted to an equivalent incremental quaternion $\Delta\mathbf{q}$ by the operations

$$\Delta\theta = |\Delta\boldsymbol{\rho}| \,(\text{rotation angle in radians}) \tag{3.33}$$

$$\mathbf{u} = \frac{1}{\theta}\Delta\boldsymbol{\rho} \tag{3.34}$$

$$= \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} (\text{unit vector}) \tag{3.35}$$

$$\Delta\mathbf{q} = \begin{bmatrix} \cos\left(\dfrac{\theta}{2}\right) \\ u_1 \sin\left(\dfrac{\theta}{2}\right) \\ u_2 \sin\left(\dfrac{\theta}{2}\right) \\ u_3 \sin\left(\dfrac{\theta}{2}\right) \end{bmatrix} \tag{3.36}$$

$$= \begin{bmatrix} \Delta q_0 \\ \Delta q_1 \\ \Delta q_2 \\ \Delta q_3 \end{bmatrix} (\text{unit quaternion}). \tag{3.37}$$

Quaternion Implementation of Attitude Integration If

$\mathbf{q}_{k-1}$ is the quaternion representing the prior value of attitude,
$\Delta\mathbf{q}$ is the quaternion representing the change in attitude, and
$\mathbf{q}_k$ is the quaternion representing the updated value of attitude,

then the update equation for quaternion representation of attitude is

$$\mathbf{q}_k = \Delta\mathbf{q} \times \mathbf{q}_{k-1} \times \Delta\mathbf{q}^*, \tag{3.38}$$

where the post superscript * represents the conjugate of a quaternion,

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}^* \overset{\text{def}}{=} \begin{bmatrix} q_1 \\ -q_2 \\ -q_3 \\ -q_4 \end{bmatrix}. \tag{3.39}$$

3.7.5.5 Direction Cosines Implementation The coordinate transformation matrix $\mathbf{C}_{\text{inertial}}^{\text{body}}$ from body-fixed coordinates to inertial coordinates is needed for transforming discretized velocity changes measured by accelerometers into inertial coordinates for integration. The quaternion representation of attitude is used for computing $\mathbf{C}_{\text{inertial}}^{\text{body}}$.

Quaternions to Direction Cosines Matrices The direction cosines matrix $\mathbf{C}_{\text{inertial}}^{\text{body}}$ from body-fixed coordinates to inertial coordinates can be computed from its equivalent unit quaternion representation,

$$\mathbf{q}_{\text{inertial}}^{\text{body}} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}, \tag{3.40}$$

as

$$\mathbf{C}_{\text{inertial}}^{\text{body}} = \left(2q_0^2 - 1\right)\mathbf{I}_3 + 2\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \times \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}^T - 2q_0 \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \otimes \tag{3.41}$$

$$= \begin{bmatrix} \left(2q_0^2 - 1 + 2q_1^2\right) & \left(2q_1q_2 + 2q_0q_3\right) & \left(2q_1q_3 - 2q_0q_2\right) \\ \left(2q_1q_2 - 2q_0q_3\right) & \left(2q_0^2 - 1 + 2q_2^2\right) & \left(2q_2^2 + 2q_0q_1\right) \\ \left(2q_1q_3 + 2q_0q_2\right) & \left(2q_2^2 - 2q_0q_1\right) & \left(2q_0^2 - 1 + 2q_3^2\right) \end{bmatrix}. \tag{3.42}$$

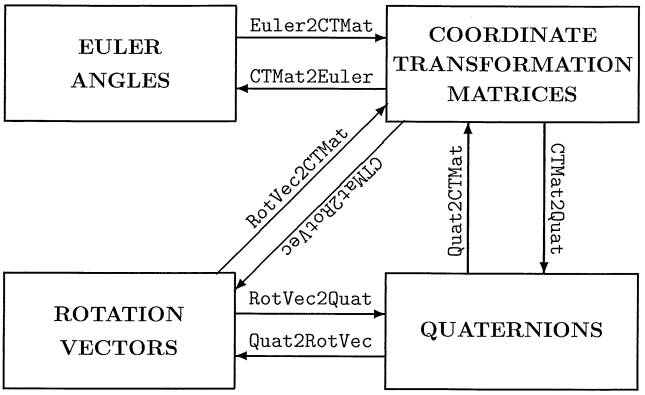


Fig. 3.20 Attitude representation formats and MATLAB® transformations.

3.7.5.6 MATLAB® Implementations The diagram in Fig. 3.20 shows four different representations used for relative attitudes and the names of the MATLAB® script m-files (i.e., with the added ending .m) on the accompanying website for transforming from one representation to another.

3.7.6 Strapdown Navigation Implementation

The basic signal processing functions for strapdown INS navigation are diagrammed in Fig. 3.21, where the common symbols used in Fig. 3.15 have the same meaning as before, and

- $\mathbf{G}$ is the estimated gravitational acceleration, computed as a function of estimated position.
- POS_{NAV} is the estimated position of the host vehicle in navigation coordinates.
- VEL_{NAV} is the estimated velocity of the host vehicle in navigation coordinates.
- ACC_{NAV} is the estimated acceleration of the host vehicle in navigation coordinates, which may be used for trajectory control (i.e., vehicle guidance).
- $\text{ACC}_{\text{SENSOR}}$ is the estimated acceleration of the host vehicle in sensor-fixed coordinates, which may be used for steering stabilization and control.
- $\mathbf{C}_{\text{NAV}}^{\text{SENSOR}}$ is the 3×3 coordinate transformation matrix from sensor-fixed coordinates to navigation coordinates, representing the attitude of the sensors in navigation coordinates.
- Ω_{SENSOR} is the estimated angular velocity of the host vehicle in sensor-fixed coordinates, which may be used for vehicle attitude stabilization and control.

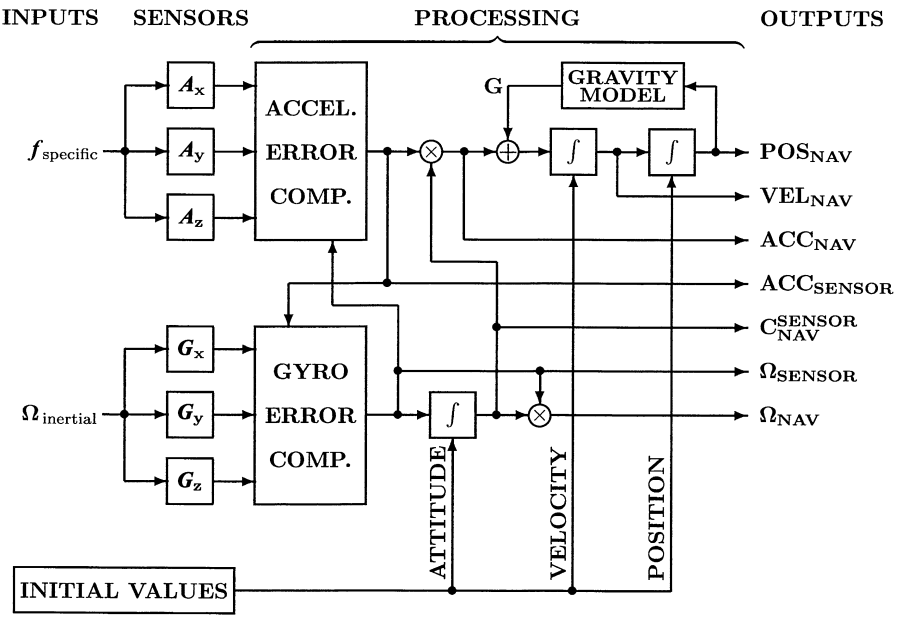


Fig. 3.21 Essential navigation signal processing for strapdown INS.

Ω_{NAV} is the estimated angular velocity of the host vehicle in navigation coordinates, which may be used in a vehicle pointing and attitude control loop.

The essential processing functions include double integration (represented by boxes containing integration symbols) of acceleration to obtain position, and computation of (unsensed) gravitational acceleration as a function of position. The sensed angular rates also need to be integrated to maintain the knowledge of sensor attitudes. The initial values of all the integrals (i.e., position, velocity, and attitude) must also be known before integration can begin.

The position vector POS_{NAV} is the essential navigation solution. The other outputs shown are not needed for all applications, but most of them (except Ω_{NAV}) are intermediate results that are available "for free" (i.e., without requiring further processing). The velocity vector VEL_{NAV}, for example, characterizes speed and heading, which are also useful for correcting the course of the host vehicle to bring it to a desired location. Most of the other outputs shown would be required for implementing control of an unmanned or autonomous host vehicle to follow a desired trajectory and/or to bring the host vehicle to a desired final position.

Navigation functions that are not shown in Fig. 3.21 include

1. How initialization of the integrals for position, velocity, and attitude is implemented. Initial position and velocity can be input from other

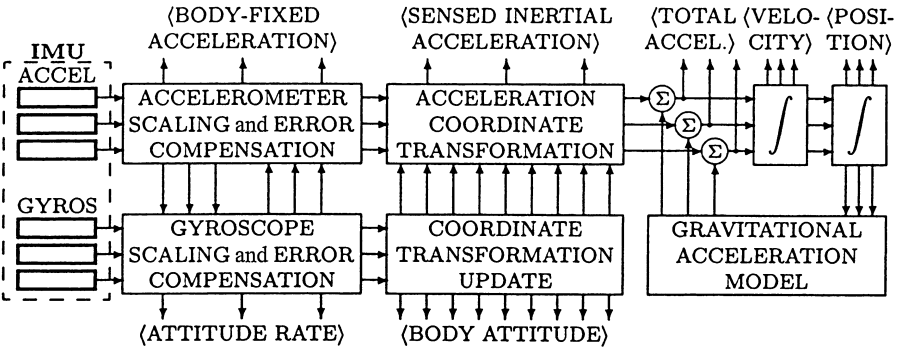


Fig. 3.22 Outputs (in angular brackets) of simple strapdown INS.

sources (e.g., GNSS), and attitude can be inferred from some form of trajectory matching (e.g., using GNSS) or by **gyrocompassing**.

2. How attitude rates are integrated to obtain attitude, described in Section 3.7.5.
3. For the case that navigation coordinates are earth-fixed, the computation of navigational coordinate rotation due to earthrate as a function of position, and its summation with sensed rates before integration.
4. For the case that navigation coordinates are locally level, the computation of the rotation rate of navigation coordinates due to vehicle horizontal velocity, and its summation with sensed rates before integration.
5. Calibration of the sensors for error compensation. If the errors are sufficiently stable, it needs to be done only once. Otherwise, it can be implemented using GNSS/INS integration techniques (Chapter 12).

Figure 3.22 is a process flow diagram for the same implementation, arranged such that the variables available for other functions is around the periphery. These are the sorts of variables that might be needed for driving GNSS phase-tracking, cockpit displays, antennas, weaponry, sensors, or other surveillance assets.

3.7.7 Navigation Computer and Software Requirements

It is not a problem today, but throughout the early history of INS development, the pace of computer development had been a limiting factor in INS development. There was no computer industry until the 1950s, no flight-qualified computers until the 1960s, and no suitable microprocessor technology until the 1970s.

The following subheadings list some of the requirements placed on navigation computers and software that tend to set them apart.

3.7.7.1 Physical and Operational Requirements These include

1. size, weight, form factor, available input power;
2. environmental conditions such as shock/vibration, temperature, electromagnetic interference (EMI);
3. memory (how much and how fast), throughput (operations/s), word length/precision;
4. time required between power-on and full operation, and minimum time between turn-off and turn-on (e.g., some vehicles shut down all power during fueling);
5. reliability, shelf life and storage requirements;
6. operating life; some applications (e.g., missiles) have operating lifetimes of minutes or seconds, others (e.g., military and commercial aircraft) may operate nearly continuously for decades; and
7. additional application-specific requirements, such as radiation hardening or the ability to function during high dynamic loading.

Most of these and their associated system interfaces have largely been standardized today, especially for military applications.

3.7.7.2 Operating Systems Inertial navigation is a *real-time* process. The tasks of sampling the sensor outputs and of integrating attitude rates, accelerations, and velocities must be scheduled at precise time intervals, and the results must be available after limited delay times. The top-level operating system that prioritizes and schedules these and other tasks must be a *real-time operating system* (RTOS). It may also be required to communicate with other computers in various ways.

3.7.7.3 Interface Requirements These include not only the operational interfaces to sensors and displays but may also include communications interfaces and specialized computer interfaces to support navigation software development and verification.

3.7.7.4 Software Development Because INS failures could put host vehicle crews and passengers at risk, it is very important during system development to demonstrate high reliability of the software. Ada is often used as the programming language because it has many built-in features to assure compliance. INS software is usually developed offline on a general-purpose computer interfaced to the navigation computer. Software development environments for INS typically include code editors, cross compilers, navigation computer emulators, hardware simulators, hardware-in-the-loop interfaces and specialized source-code-online interfaces to the navigation computer for monitoring, debugging and verifying the navigation software on the navigation computer.

Software developed for manned missions must be acceptably reliable, which requires metrics for demonstrating reliability and testing for verification.

3.8 INS PERFORMANCE STANDARDS

3.8.1 Free Inertial Operation

Operation of an INS without external aiding of any sort is called **free inertial** or **pure inertial**. Because free inertial navigation in the near-earth gravitational environment is unstable in the vertical direction (due to the falloff of gravity with increasing altitude), aiding by other sensors (e.g., barometric altimeters for aircraft or surface vehicles, radar altimeters for aircraft over water, or hydrostatic pressure for submersibles) is required to avoid vertical error instability. For that reason, performance of free INSs is usually specified for horizontal position errors only.

3.8.2 INS Performance Metrics

Oversimplified error model: INS position is initialized by knowing where you are starting from at initial time t_0. The position error may start out very small, but it tends to increase with time due to the influence of sensor errors. Double integration of accelerometer output errors is a major source of this growth over time. Experience has shown that the variance and standard deviation of horizontal position error,

$$\sigma^2_{\text{position}}(t) \propto (t - t_0)^2 \tag{3.43}$$

$$\sigma_{\text{position}}(t) \approx C \times |t - t_0|, \tag{3.44}$$

with unknown positive constant C. This constant C would then characterize performance of an INS in terms of how fast its RMS position error grows.

A problem with this model is that actual horizontal INS position errors are two-dimensional, and we would need a **2×2 covariance matrix** in place of C. That would not be very useful in practice. As an alternative, we replace C with something more intuitive and practical.

CEP is the radius of a horizontal circle centered at the estimated position, and of sufficient radius such that it is equally probable that the true horizontal position is inside or outside the circle. CEP is the acronym for either **circular error probable** or for **circle of equal probability**, depending on which is easier to remember.

CEP rate is the time rate of change of CEP. Traditional units of CEP rate are **nautical miles per hour** or **kilometers per hour**. The nautical mile was

originally intended to designate a surface distance equivalent to 1 arc minute of latitude change at sea level, but that depends on latitude. The Système International (SI)-derived nautical mile is 1.852 km.

3.8.3 Performance Standards

In the 1970s, before GPS became a reality, the U.S. Air Force had established the following standard levels of performance for INS:

High-accuracy systems have free inertial CEP rates in the order of 0.1 nmi/h (≈**185** m/h) or better. This is the order of magnitude in accuracy required for intercontinental ballistic missiles (ICBMs) and missile-carrying submarines, for example.

Medium-accuracy systems have free inertial CEP rates in the order of 1 nmi/h (≈**1.85** km/h). This is the level of accuracy deemed sufficient for most military and commercial aircraft [2].

Low-accuracy systems have free inertial CEP rates in the order of 10 nmi/h (≈**18.5** km/h) or worse. This range covered the requirements for many short-range standoff weapons such as guided artillery or tactical rockets.

Table 3.2 lists accelerometer and gyroscope performance ranges compatible with the standards.

However, after GPS became available, GPS/INS integration could make a low-accuracy INS behave more like a high-accuracy INS.

3.9 TESTING AND EVALUATION

The final stage of the development cycle is testing and performance evaluation. For stand-alone inertial systems, this usually proceeds from the laboratory to a succession of host vehicles, depending on the application.

3.9.1 Laboratory Testing

Laboratory testing is used to evaluate sensors before and after their installation in the ISA, and then to evaluate the system implementation during

TABLE 3.2. INS and Inertial Sensor Performance Ranges

	Performance Ranges			
System or Sensor	High	Medium	Low	Units
INS	$\leq 10^{-1}$	≈ 1	≥ 10	nmi/h[a]
Gyroscopes	$\leq 10^{-3}$	$\approx 10^{-2}$	$\geq 10^{-1}$	deg/h
Accelerometers	$\leq 10^{-7}$	$\approx 10^{-6}$	$\geq 10^{-5}$	g (9.8 m/s^2)

[a]Nautical miles per hour CEP rate.

operation. The navigation solution from a stationary system should remain stationary, and any deviation is due to navigation errors. Testing with the system stationary can also be used to verify that position errors due to intentional initial velocity errors follow a path predicted by Schuler oscillations (≈**84.4**-min period, described in Chapter 11) and the Coriolis effect. If not, there is an implementation error. Other laboratory testing may include controlled tilts and rotations to verify the attitude estimation implementations, and detect any uncompensated sensitivities to rotation and acceleration.

Additional laboratory testing may be required for specific applications. Systems designed to operate aboard Navy ships, for example, may be required to meet their performance requirements under dynamic disturbances at least as bad as those to be expected aboard ships under the worst sea conditions. This may include what is known as a "Scoresby test," used at the U.S. Naval Observatory in the early twentieth century for testing gyrocompasses. Test conditions may include roll angles of **±80°** and pitch angles of **±15°**, at varying periods in the order of a second.

Drop tests (for survival testing) and shake-table or centrifuge tests (for assessing acceleration capabilities) can also be done in the laboratory.

3.9.2 Field Testing

After laboratory testing, systems are commonly evaluated next in highway testing.

Systems designed for tactical aircraft must be designed to meet their performance specifications under the expected peak dynamic loading, which is generally determined by the pilot's limitations.

Systems designed for rockets must be tested under conditions expected during launch, sometimes as a "piggyback" payload during the launch of a rocket for another purpose. Accelerations can reach around 3 g for manned launch vehicles, and much higher for unmanned launch vehicles.

In all cases, GNSS has become an important part of field instrumentation. The Central Inertial Guidance Test Facility (CIGTF) at Holoman AFB has elaborate range instrumentation for this purpose. This facility is used by NASA and Department of Defense (DoD) agencies for INS and GNSS/INS testing in a range of host vehicles.

3.10 SUMMARY

1. Inertial navigation has a rich technology base—more than can be covered in a single book, and certainly not in one chapter. Unfortunately, some of best technology is either classified or proprietary. Otherwise, there is some good open-source literature:
   (a) Titterton and Weston [9] is a good source for additional information on strapdown hardware and software.

   (b) Paul Savage's two-volume tome [8] on strapdown system implementations is also rather thorough.
   (c) Chapter 5 of Ref. 3 and the references therein include some recent developments.
   (d) Journals of the Institute of Electrical and Electronics Engineers (IEEE), Institution of Electrical Engineers (IEE), Institute of Navigation, and other professional engineering societies generally have the latest developments on inertial sensors and systems.
   (e) In addition, the World Wide Web includes many surveys and reports on inertial sensors and systems.
2. Inertial navigation accuracy is mostly limited by inertial sensor accuracy.
3. The accuracy requirements for inertial sensors cannot always be met within manufacturing tolerances. Some form of calibration is usually required for compensating the residual errors.
4. INS accuracy degrades over time, and the most accurate systems generally have the shortest mission times. For example, ICBMs only need their inertial systems for a few minutes.
5. Performance of inertial systems is commonly specified in terms of CEP rate.
6. Accelerometers cannot measure gravitational acceleration.
7. Both inertial and satellite navigation require accurate models of the earth's gravitational field.
8. Both navigation modes also require an accurate model of the shape of the earth.
9. The first successful navigation systems were gimbaled, in part because the computer technology required for strapdown implementations was decades away. That has not been a problem for about four decades.
10. Gimbaled systems tend to be more accurate and more expensive than strapdown systems.
11. The more reliable attitude implementations for strapdown systems use quaternions to represent attitude.
12. Systems traditionally go through a testing and evaluation process to verify performance.
13. Before testing and evaluation of an INS, its expected performance is commonly evaluated using the analytical models of Chapter 11.

PROBLEMS

Refer to Appendix B for coordinate system definitions and satellite orbit equations.

3.1 Which, if any, of the following coordinate systems is not rotating?

(a) NED

(b) ENU

(c) ECEF

(d) ECI

(e) Moon-centered, moon-fixed.

3.2 What is the minimum number of two-axis gyroscopes (i.e., gyroscopes with two, independent, orthogonal input axes) required for inertial navigation?

(a) 1

(b) 2

(c) 3

(d) Not determined.

3.3 What is the minimum number of gimbal axes required for gimbaled inertial navigators in fully maneuverable host vehicles? Explain your answer.

(a) 1

(b) 2

(c) 3

(d) 4

3.4 Define specific force.

3.5 An ISA operating at a fixed location on the surface of the earth would measure

(a) no acceleration

(b) 1 g acceleration downward

(c) 1 g acceleration upward.

3.6 Explain why an INS is not a good altimeter.

3.7 The inertial rotation rate of the earth is

(a) 1 revolution per day

(b) 15 deg/h

(c) 15 arc seconds/s

(d) ≈15.0411 arc seconds/s.

3.8 Define the CEP and the CEP rate for an INS.

3.9 The CEP rate for a medium-accuracy INS is in the order of

(a) 2 m/s

(b) 200 m/h

(c) 2000 m/h

(d) 20 km/h

3.10 In the one-dimensional line land world of Section 3.7.1, an INS requires no gyroscopes. How many gyroscopes would be required for two-dimensional navigation in flat land?

3.11 Derive the equivalent formulas in terms of Y (yaw angle), P (pitch angle), and R (roll angle) for unit vectors 1_R, 1_P, 1_Y in NED coordinates and 1_N, 1_E, 1_D in RPY coordinates.

3.12 Explain why accelerometers cannot sense gravitational accelerations.

3.13 Show that the matrix $\mathbf{C}_{\text{inertial}}^{\text{body}}$ defined in Eq. 3.42 is orthogonal by showing that $\mathbf{C}_{\text{inertial}}^{\text{body}} \times \mathbf{C}_{\text{inertial}}^{\text{body T}} = \mathbf{I}$, the identity matrix. (Hint: Use $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$.)

3.14 Calculate the numbers of computer multiplies and adds required for

(a) gyroscope scale factor/misalignment/bias compensation (Eq. 3.4 with $N = 3$),

(b) accelerometer scale factor/misalignment/bias compensation (Eq. 3.4 with $N = 3$), and

(c) transformation of accelerations to navigation coordinates (Fig. 3.22) using quaternion rotations (see Appendix B section on quaternion algebra).

If the INS performs these 100 times per second, how many operations per second will be required?

REFERENCES

[1] J. E. Bortz, "A New Mathematical Formulation for Strapdown Inertial Navigation," *IEEE Transactions on Aerospace and Electronic Systems* **AES-6**, 61–66 (1971).

[2] Chairman of Joint Chiefs of Staff, U.S. Department of Defense, *2003 CJCS Master Positioning, Navigation and Timing Plan*, Report CJCSI 6130.01C, March 2003.

[3] P. D. Groves, *Principles of GNSS, Inertialm and Multisensor Integrated Navigation Systems*. Artech House, London, 2008.

[4] *IEEE Standard for Inertial Sensor Terminology*, IEEE Standard 528-2001, Institute of Electrical and Electronics Engineers, New York, 2001.

[5] *IEEE Standard for Inertial System Terminology*, IEEE Standard 1559, Institute of Electrical and Electronics Engineers, New York, 2007.

[6] D. Mackenzie, *Inventing Accuracy: A Historical Sociology of Nuclear Missile Guidance*. MIT Press, Cambridge, MA, 1990.

[7] F. K. Mueller, "A History of Inertial Navigation," *Journal of the British Interplanetary Society* **38**, 180–192 (1985).

[8] P. G. Savage, *Introduction to Strapdown Inertial Navigation Systems*, Vols. 1 & 2. Strapdown Associates, Maple Plain, MN, 1996.

[9] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*. IEE, London, 2004.

4

GNSS SIGNAL STRUCTURE, CHARACTERISTICS, AND INFORMATION UTILIZATION

Why are the global navigation satellite system (GNSS) signals so complex? GNSSs are designed to be readily accessible to millions of military and civilian users. The GNSSs are receive-only passive systems that enable a very large number of users to simultaneously use the system. Because there are many functions that must be performed, the GNSS signals have a rather complex structure. As a consequence, there is a correspondingly complex sequence of operations that a GNSS receiver must carry out in order to extract and utilize the desired information from the signal. Modernized GPS and other GNSSs have improved upon the legacy GPS waveform in terms of multipath, correlation, and overall performance. In this chapter, we characterize the signal mathematically, describe the purposes and properties of the important signal components, and discuss generic methods for extracting information from these GNSS navigation signals.

This chapter will begin with a presentation of the legacy GPS signals, followed by the modernized GPS signals. Discussions of the Global Orbiting Navigation Satellite System (GLONASS), Galileo, Compass/BeiDou (BD), and the Quasi-Zenith Satellite System (QZSS) follow.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

4.1 LEGACY GPS SIGNAL COMPONENTS, PURPOSES, AND PROPERTIES

The GPS is a code-division multiple access (CDMA) satellite-based ranging system. The system is considered a spread-spectrum system where the radio frequency (RF) bandwidth that is used is much wider than as required to transmit the underlying navigation data. The GPS has two basic services that are provided by the legacy GPS: the Standard Positioning Service (SPS) [1] and the Precise Positioning Service (PPS).

The legacy GPS signals in space are well documented by the U.S. Department of Defense in the form of an interface specification (IS). This IS provides basic characteristics of the signal in space and information on how user equipment should interface to and process the navigation signals [2].

The interface between the GPS space and user segments consists of two RF links, L1 and L2. The carriers of the L-band links can be modulated by two navigation data bit streams, each of which normally is a composite generated by the modulo-2 addition of a pseudorandom noise (PRN) ranging code and the downlink system navigation data. Utilizing these links, the space vehicles (SVs) of the GPS space segment can provide continuous Earth coverage of navigation signals that provide the user segment the ranging codes and system data needed for the user to calculate a position, velocity, and time solution. These signals are available to a suitably equipped GPS user with RF visibility to the SVs.

4.1.1 Mathematical Signal Models for the Legacy GPS Signals

Each GPS satellite simultaneously transmits on two of the legacy L-band frequencies denoted by L1 and L2, which are 1575.42 and 1227.60 MHz, respectively. The carrier of the L1 signal consists of two signal components that are orthogonal (i.e., cos and sin functions in quadrature). The first component is biphase modulated (i.e., binary phase shift keying [BPSK]) by a 50-bps (bits per second) navigation data stream and a PRN spreading code, called the *C/A-code*, consisting of a 1023-chip sequence that has a period of 1 ms and a chipping rate of 1.023 MHz. The second component is also biphase modulated by the same 50-bps data stream but with a different PRN code called the *P(Y)-code*, which has a 10.23-MHz chipping rate and has a 1-week period. While the native P-code can be transmitted without data encryption, when the P-code is encrypted, it is referred to as the P(Y)-code. The mathematical model of the GPS navigation signal on the L1 frequency is shown in Eq. 4.1, for each SV i:

$$s_i(t) = \sqrt{2P_{\mathrm{C/A}}}\, d(t) c_i(t) \cos(\omega t + \theta_i) + \sqrt{2P_{\mathrm{P(Y)}}}\, d(t) p_i(t) \sin(\omega t + \theta_i), \quad (4.1)$$

where

$P_{C/A}$ = power in C/A-encoded component
$P_{P(Y)}$ = power in P(Y)-encoded component
$d(t)$ = navigation date at 50-bps rate
$c_i(t)$ = C/A PRN code at 1.023-MHz rate for SV i
$p_i(t)$ = P PRN code at 10.23-MHz rate for SV i
ω = carrier frequency (rad/s)

In Eq. 4.1, $P_{C/A}$ and $P_{P(Y)}$ are the respective carrier powers for the C/A- and P(Y)-encoded carrier components. The data, $d(t)$, is the 50-bps navigation data modulation; $c(t)$ and $p(t)$ are the respective C/A and P PRN code waveforms; ω is the L1 carrier frequency in radian per second; and θ is a common phase shift in radians at the respective carrier frequency. The C/A-code carrier component lags the P-code carrier component by 90° when both data chips are 0 (i.e., –1). The carrier power for the P-code carrier is approximately 3 dB less than the power in the C/A-code carrier [2].

In contrast to the L1 signal, the L2 signal is modulated with only the navigation data at the 50-bps rate and the P(Y)-code, although there is the option of not transmitting the 50-bps data stream. The mathematical model of the L2 waveform is shown in Eq. 4.2 for each SV i:

$$s_i(t) = \sqrt{2P_{P(Y)}}\, d(t) p_i(t) \sin(\omega t + \theta_i), \tag{4.2}$$

where

$P_{P(Y)}$ = power in P(Y)-encoded component
$d(t)$ = navigation date at 50-bps rate
$p_i(t)$ = P PRN code at 10.23-MHz rate for SV i
ω = carrier frequency (rad/s).

Figure 4.1 illustrates a function block diagram of how the legacy GPS signals are generated at L1 and L2. Both of the output signals go to their respective power amplifiers and are then combined for transmission out of the GPS helix antenna array. The gain adjustments depicted in Fig. 4.1 as negative values would be implemented in the generation of the signals themselves and not as actual attenuation.

It is worthy to note that all signals generated are from the same reference oscillator, and this reference oscillator is divided by integer numbers for the C/A-code clock, P(Y)-code clock, navigation data register clock, and multiplied up by an integer number for the L1 and L2 carrier signals. This helps in the synthesizer and clock designs as well as in maintaining code to carrier coherency.

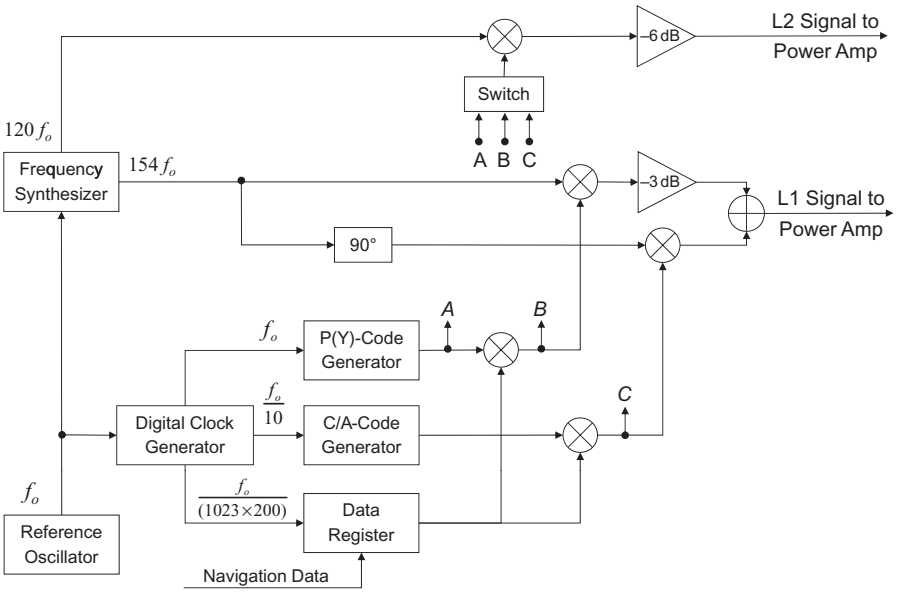


Fig. 4.1 Block diagram of GPS signal generation at L1 and L2 frequencies. *Note*: All signals coherent with reference oscillator.

Figure 4.2 illustrated the C/A-encoded structure on the L1 carrier signal. The 50-bps navigation data bit boundaries always occur at an epoch of the C/A-code. The C/A-code epochs mark the beginning of each period of the C/A-code, and there are precisely 20 code epochs per data bit, which simply repeats 20 times. Within each C/A-code chip, there are precisely 1540 L1 carrier cycles. The navigation data and code are multiplied together (implemented digitally with a modulo addition) and are then multiplied by the carrier signal. When $(d(t) \oplus c(t))$ is a "1," simple the phase of the carrier is unaffected, but when $(d(t) \oplus c(t))$ is a "–1," the phase of the carrier is inverted by 180° to produce the BPSK modulation.

Figure 4.3 illustrated the P(Y)-encoded structure on the L1 carrier, with the navigation data bits. This modulation process is similar to the C/A signal encoding with some key differences. The carrier competent for the P(Y)-code signal is in quadrature (i.e., is orthogonal) to that used for the C/A-code generation. Also, the P(Y) is more random and does not repeat within a navigation data bit interval, and there will be a total of 204,600 P(Y)-code chips within a navigation data bit of duration 20 ms; again, the P(Y) is in alignment with the data bit boundaries. The P(Y)-code runs at a 10 times faster rate than that of the C/A-code. Hence, there will be less carrier cycles per chip in the final signal. For the GPS L1 P(Y) signal, there are exactly 154 L1 carrier cycles per P(Y) chip. BPSK modulation is again performed.

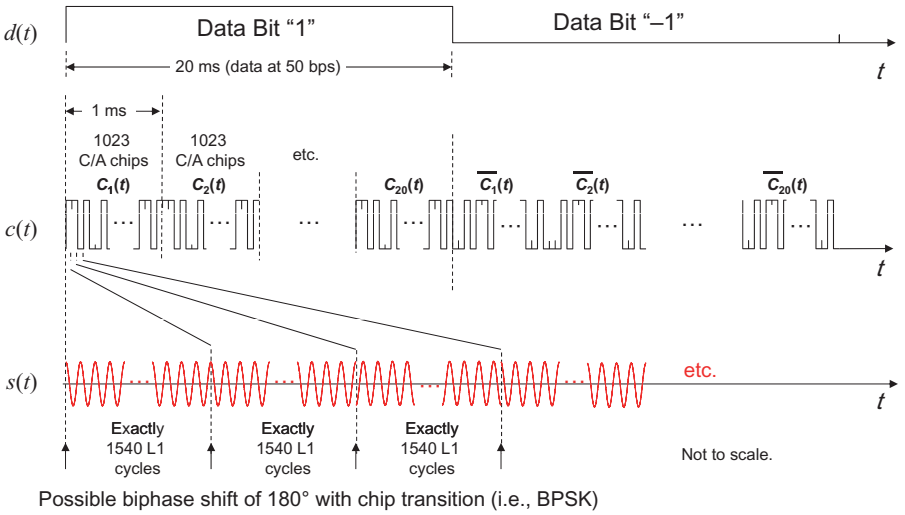


Fig. 4.2 GPS C/A signal structure at L1 generation illustration.

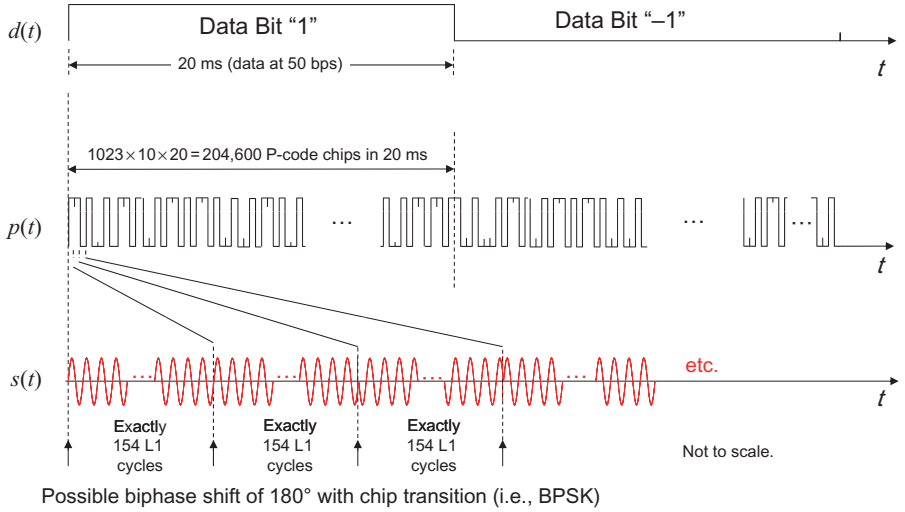


Fig. 4.3 GPS P(Y) signal structure at L1 generation illustration.

When the C/A-encoded signal is combined with the P(Y)-encoded signal, there are two, synchronized BPSK signals transmitted in quadrature. This feature allows these two signals to be separated in the GPS receiver.

4.1.2 Navigation Data Format

The navigation data bit stream running at a 50-bps rate conveys the *navigation message* that is determined by the GPS Ground Control Segment and uploaded

to each satellite. This navigation information is then encoded onto the navigation signals and transmitted to the users. The navigation message includes, but is not limited to, the following information:

1. *Satellite Almanac Data.* Each satellite transmits orbital data called the *almanac*, which enables the user to calculate the approximate location of every satellite in the GPS constellation at any given time. Almanac data are not accurate enough to determine an accurate user position but can be stored in a receiver where they remain valid for many months, for their intended purpose. They are used primarily to determine which satellites are visible at an estimated location so that the receiver can search for those satellites when it is first turned on. They can also be used to determine the approximate expected signal Doppler shift to aid in rapid acquisition of the satellite signals. Every GPS satellite transmits almanac data about the GPS constellation (i.e., about all the other satellites, including itself.) Almanac data include the satellite orbital (i.e., position) and satellite clock error course data.
2. *Satellite Ephemeris Data.* Ephemeris data are similar to almanac data but enable a much more accurate determination of satellite position needed to convert signal propagation delay into an estimate of user position. Satellite ephemeris data include details on the position of the satellite (i.e., orbital ephemeris data) and details on the satellite's clock error (i.e., clock ephemeris data). In contrast to almanac data, ephemeris data for a particular satellite are broadcast only by that satellite and the data are typically valid for many hours (e.g., 0–24 hrs).
3. *Signal Timing Data.* The navigation data stream includes time tagging, which is used to establish the transmission time of specific points on the GPS signal. This information is needed to determine the satellite-to-user propagation delay used for ranging.
4. *Ionosphere Delay Data.* Ranging errors due to the signal propagation delay through the earth's ionosphere can affect GPS positioning. While these effects can be measured directly for dual-frequency GPS users, they must be modeled for single-frequency users. To help mitigate these effects, ionosphere error perditions are encoded into the navigation message that can be decoded and applied by the user to partially cancel these errors.
5. *Satellite Health Message.* The navigation data stream also contains information regarding the current health of the satellite, so that the receiver can ignore that satellite if it is not operating properly.

The format of the navigation message has a basic frame structure as shown in Fig. 4.4. The format consists of 25 frames, where a frame contains 1500 bits. Each *frame* is subdivided into five 300-bit subframes, and each *subframe* consists of 10 words of 30 bits each, with the most significant bit (MSB) of the word transmitted first. Thus, at the 50-bps rate, it takes 6 s to transmit a subframe and 30 s to complete one frame. Transmission of the complete 25-frame

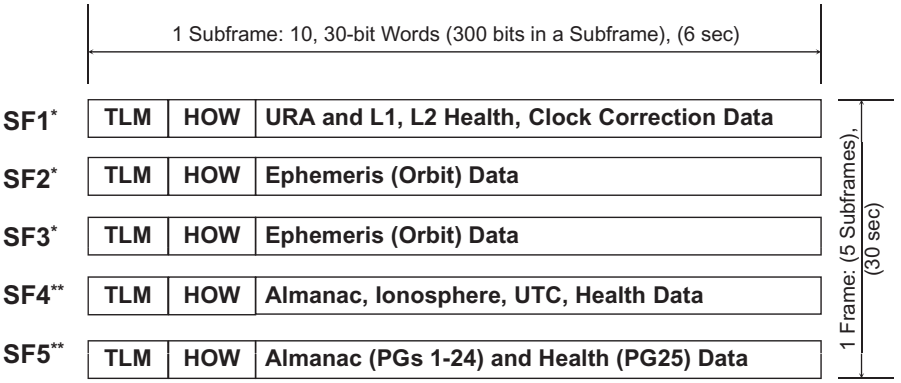


Fig. 4.4 GPS navigation data format (legacy) frame structure. *Same data transmitted every frame, by each SV (until updated by CS). **Data commutated in each frame (i.e., page), common to all SVs (until updated by CS). *Note*: With multichannel receiver, data are decoded simultaneously from SVs.

navigation message requires 750 s, or 12.5 min. Except for occasional updating, subframes 1, 2, and 3 are constant (i.e., repeat) with each frame at the 30-s frame repetition rate. On the other hand, subframes 4 and 5 are each subcommutated 25 times. The 25 versions of subframes 4 and 5 are referred to as *pages* 1–25. Hence, except for occasional updating, each of these pages repeats every 750 s, or 12.5 min.

A detailed description of all information contained in the navigation message is beyond the scope of this text but can be found in Ref. 2. Therefore, we give only an overview of the fundamental elements. Each subframe begins with a *telemetry word* (TLM). The first 8 bits of the TLM is a preamble that enables the receiver to determine when a subframe begins and for data recovery purposes. The remainder of the TLM contains parity bits and a telemetry message that is available only to authorized users. The second word of each subframe is called the *handover word* (HOW).

4.1.2.1 Z-Count Information contained in the HOW is derived from a 29-bit quantity called the *Z-count*. The *Z*-count is not transmitted as a single word, but part of it is transmitted within the HOW. The *Z*-count counts *epochs* generated by the X1 register of the P-code generator in the satellite, which occur every 1.5 s.

The 19 least significant bits (LSBs) of the *Z*-count is called the *time of week* (TOW) count and indicate the number of X1 epochs that have occurred since the start of the current week. The start of the current week occurs at the X1 epoch, which occurs at approximately midnight of Saturday night/Sunday morning. The TOW count increases from 0 at the start of the week to 403,199 and then resets to 0 again at the start of the following week. A TOW count of

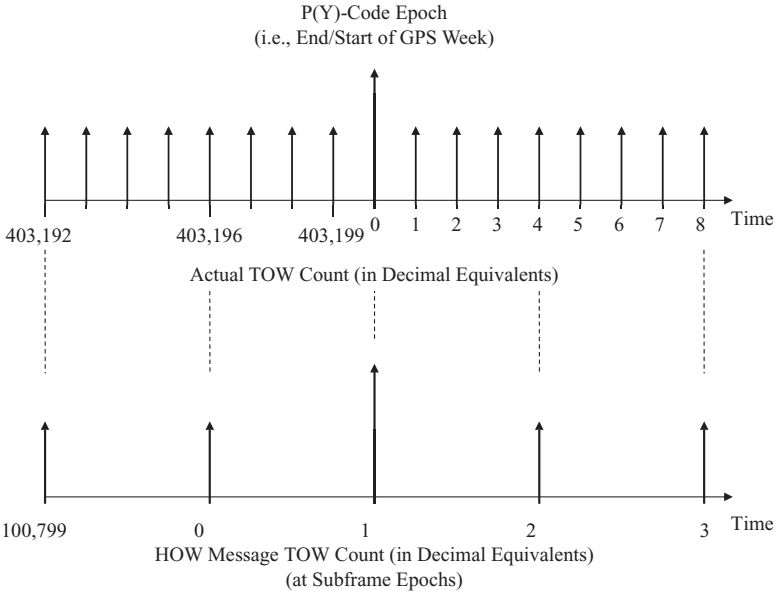


Fig. 4.5 Relationship between GPS HOW counts and TOW counts [2].

0 always occurs at the beginning of subframe 1 of the first frame (the frame containing page 1 of subcommutated subframes 4 and 5) [2].

A truncated version of the TOW count, containing its 17 MSBs, constitutes the first 17 bits of the HOW. Multiplication of this truncated count by 4 gives the TOW count at the start of the following subframe. Since the receiver can use the TLM preamble to determine precisely the time at which each subframe begins, a method for determining the time of transmission of any part of the GPS signal is thereby provided. The time projection to the next subframe beginning can also be used to rapidly acquire the P(Y)-code, which has a week-long period. The relationship between the HOW counts and TOW counts is shown in Fig. 4.5.

4.1.2.2 GPS Week Number (WN) The 10 MSBs of the *Z*-count contain the GPS *week number* (WN), which is a modulo-1024 week count. The *zero state* is defined to be the week that started with the X1 epoch occurring at approximately midnight on the night of January 5, 1980/morning of January 6, 1980, where GPS time began. Because WN is a modulo-1024 count, an event called the *week rollover* occurs every 1024 weeks (a few months short of 20 years), and GPS receivers must be designed to accommodate. The last GPS WN rollover occurred at GPS time zero on August 22, 1999 with few difficulties. The WN is not part of the HOW but instead appears as the first 10 bits of the third word in subframe 1.

Frame and Subframe Identification Three bits of the HOW are used to identify which of the five subframes is being transmitted. The frame being transmitted (corresponding to a page number from 1 to 25) can readily be identified from the TOW count computed from the HOW of subframe 5. This TOW count is the TOW at the start of the next frame. Since there are 6 TOW counts per frame, the frame number of that frame is simply (TOW/6) (mod 25).

4.1.2.3 Information by Subframe In addition to the TLM and HOW, which occur in every subframe, the following information is contained within the remaining eight words of subframes 1–5 (only fundamental information is described):

1. *Subframe 1.* The WN portion of the *Z*-count is part of word 3 in this subframe. Subframe 1 also contains GPS clock correction data for the satellite in the form of polynomial coefficients defining how the correction varies with time. Time defined by the clocks in the satellite is commonly called *SV time*; the time after corrections have been applied is called *GPS time*. Thus, even though individual satellites may not have perfectly synchronized SV times, they do share a common GPS time. Additional information in subframe 1 includes the quantities t_{oc}, T_{GD}, and issue of data clock (IODC). The clock reference time, t_{oc}, is used as a time origin to calculate satellite clock error. The T_{GD} term is used to correct for satellite group delay errors for single-frequency GPS users, relative to an L1 and L2 dual-frequency P(Y) user solution. The IODC indicates the issue number of the clock data intended to alert users to changes in clock error parameters provided by the navigation message.
2. *Subframes 2 and 3.* These subframes contain the ephemeris data, which are used to determine the precise satellite position and velocity required by the user solution. Unlike the almanac data, these data are very precise, are valid over a relatively short period of time (e.g., 0–24 hours), and apply only to the satellite transmitting it. The issue of data ephemeris (IODE) informs users when changes in ephemeris parameters have occurred within the navigation data. Each time new parameters are uploaded from the GPS Ground Control Segment, the IODE number changes.
3. *Subframe 4.* The 25 pages of this subframe contain the almanac for satellites with PRN numbers 25 and higher, as well as special messages, ionosphere correction terms, and coefficients to convert GPS time to coordinated universal time (UTC) time. There are also spare words for possible future applications. The components of an almanac (for orbit and clock) are very similar to those of the ephemeris but contain less bits. The satellite positions and SV clock error can be calculated in a similar fashion as is done using the accurate ephemeris data.
4. *Subframe 5.* The 25 pages of this subframe include the almanac for satellites with PRN numbers from 1 to 24 and SV health information.

It should be noted that since each satellite transmits all 25 pages, almanac data for all satellites are transmitted by every satellite. Unlike ephemeris data, almanac data remain valid for long periods (e.g., months) but are much less precise. Additional data contained in the navigation message is the user range error (URE), which estimate the range error due to satellite ephemeris and timing errors (but no errors due to propagation) on a per SV i basis.

4.1.3 GPS Satellite Position Calculations

The precise positions of the GPS satellites are calculated by the GPS Ground Control Segment and then encode information pertaining to their positions in the navigation data bit stream, which is uploaded to the satellite and then broadcast to the user. The ephemeris parameters describe the orbit during the interval of time (at least 1 h) for which the parameters are transmitted. This representation model is characterized by a set of parameters that is an extension (including drag) to the Keplerian orbital parameters. The definitions of the parameters are given in Table 4.1.

After the GPS receiver has decoded these data, the precise positions of the satellites will be needed to enable the user solutions for position, velocity, and time determination. GPS uses an earth-centered, earth-fixed (ECEF) coordinate system with respect to a World Geodetic System 1984 (WGS84) datum using GPS time as established by the U.S. Naval Observatory. The SV positions and hence the user solutions will be with respect to these system references.

TABLE 4.1. Components of Ephemeris Data

Term	Description	Unit
t_{0e}	Reference time of ephemeris	Second
$\sqrt{a}$	Square root of semimajor axis	$\sqrt{\text{meter}}$
e	Eccentricity	Dimensionless
i_0	Inclination angle (at time t_{0e})	Semicircle
Ω_0	Longitude of the ascending node (at weekly epoch)	Semicircle
ω	Argument of perigee (at time t_{0e})	Semicircle
M_0	Mean anomaly (at time t_{0e})	Semicircle
$IDOT$	Rate of change of inclination angle (i.e., di/dt)	Semicircle/s
$\dot{\Omega}$	Rate of change of longitude of the ascending node	Semicircle/s
Δn	Mean motion correction	Semicircle/s
C_{uc}	Amplitude of cosine correction to argument of latitude	Radian
C_{us}	Amplitude of sine correction to argument of latitude	Radian
C_{rc}	Amplitude of cosine correction to orbital radius	Meter
C_{rs}	Amplitude of sine correction to orbital radius	Meter
C_{ic}	Amplitude of cosine correction to inclination angle	Radian
C_{is}	Amplitude of sine correction to inclination angle	Radian

TABLE 4.2. Algorithm for Computing Satellite Position

	$\mu = 3.986005 \times 10^{14}$ m^3/s^2	WGS84 value of Earth's universal gravitational parameter
	$c = 2.99792458 \times 10^8$ m/s	GPS value for speed of light
	$\dot{\Omega}_e = 7.2921151467 \times 10^{-5}$ rad / s	WGS84 value of Earth's rotation rate
	$\pi = 3.1415926535898$	GPS value for ratio of circumference to radius of circle
1	$a = \left(\sqrt{a}\right)^2$	Semimajor axis
2	$n = \sqrt{\dfrac{\mu}{a^3}} + \Delta n$	Corrected mean motion (rad/s)
3	$t_k = t - t_{0e}$	Time from ephemeris epoch
4	$M_k = M_0 + (n)(t_k)$	Mean anomaly
5	$M_k = E_k - e\sin E_k$	Eccentric anomaly (must be solved iteratively for E_k)
6	$\sin v_k = \dfrac{\sqrt{1-e^2}\sin E_k}{1-e\cos E_k}$ $\cos v_k = \dfrac{\cos E_k - e}{1-e\cos E_k}$ $v_k = \text{atan2}\left[\dfrac{\sin v_k}{\cos v_k}\right]$	True anomaly (solve for in each quadrant)
7	$\phi_k = v_k + \omega$	Argument of latitude
8	$\delta\phi_k = C_{us}\sin(2\phi_k) + C_{uc}\cos(2\phi_k)$	Argument of latitude correction
9	$\delta r_k = C_{rs}\sin(2\phi_k) + C_{rc}\cos(2\phi_k)$	Radius correction
10	$\delta i_k = C_{is}\sin(2\phi_k) + C_{ic}\cos(2\phi_k)$	Inclination correction
11	$u_k = \phi_k + \delta\phi_k$	Corrected argument of latitude
12	$r_k = a(1 - e\cos E_k) + \delta r_k$	Corrected radius
13	$i_k = i_0 + (\text{di/dt})t_k + \delta i_k$	Corrected inclination
14	$\Omega_k = \Omega_0 + \left(\dot{\Omega} - \dot{\Omega}_e\right)(t_k) - \dot{\Omega}_e t_{0e}$	Corrected longitude of ascending node
15	$x'_k = r_k\cos u_k$	In-plane x position (ECI frame)
16	$y'_k = r_k\sin u_k$	In-plane y position (ECI frame)
17	$x_k = x'_k\cos\Omega_k - y'_k\cos i_k\sin\Omega_k$	ECEF x coordinate
18	$y_k = x'_k\sin\Omega_k + y'_k\cos i_k\cos\Omega_k$	ECEF y coordinate
19	$z_k = y'_k\sin i_k$	ECEF z coordinate

The Keplerian and other GPS system parameter information decoded from the navigation data stream can be used by the GPS user to calculate the satellite position with the algorithm outlined in Table 4.2. Overall, the user GPS receiver SV position calculation algorithm uses the data provided in the navigation signal broadcast and calculates the position of the SV in the orbital plane (with respect to an earth-centered inertial [ECI] coordinate system) and then rotates this SV position into the ECEF coordinate frame. While the long list of equations in Table 4.2 looks a bit daunting at first glance, it is fairly straightforward, but a couple of items deserve additional discussion.

Calculation of the satellite's ECEF antenna phase center position is very sensitive to small perturbations in most ephemeris parameters. The sensitivity of position to the parameters $\sqrt{a}$, C_{rc}, and C_{rs}, is about 1 m/m. The sensitivity to angular parameters is on the order of 10^8 m/semicircle and to the angular rate parameters on the order of 10^{12} m/semicircle/s. Because of this extreme sensitivity, when performing SV calculations in any GNSS system, it is very important to utilize the fixed parameters defined for the system and full precision (usually double precision) in the computation engine used. Some of these parameters are identified at the beginning of Table 4.2.

4.1.3.1 Ephemeris Data Reference Time Step and Transit Time Correction The reference time for the ephemeris orbit data, t_{0e}, represents the middle of the orbit fit, for which the ephemeris data are valid. When propagating the GPS SV position at a particular GPS time (t), the time index t_k is used. Thus, t_k can be positive or negative. With a time corrected GPS time t, the SV position will be the position of the SV at the time of transmission. Additionally, t_k must account for beginning- or end-of-week crossovers. Thus, if t_k is greater than 302,400 s, subtract 604,800 s from t_k; if t_k is less than –302,400 s, add 604,800 s to t_k.

An additional compensation that must be made by the user to provide an accurate user solution is compensation for the transit time ($t_{transit}$), which is the finite amount of time it takes the SV signal to propagate from the satellite antenna to the user antenna. First, the transit time must be estimated. This can be done by the GPS receiver using the adjusted pseudorange measurement (removing the transmitter clock error and T_{GD} for single frequency solution) and dividing by the speed of light. Propagation times are typically 68–83 ms, depending upon the aspect angle to the satellite. With the transit time, SV position compensations can be done by subtracting an additional rotation term to the corrected longitude of ascending node term of Table 4.2 by the amount of $\dot{\Omega}_e t_{transit}$, to compensate for the transmit time.

4.1.3.2 True, Eccentric, and Mean Anomaly With the GPS satellite in a "circular orbit," they are in reality in a slightly elliptical orbit that cannot be ignored. Orbit phase variables used for determining the position of a satellite in its elliptical orbit are illustrated in Fig. 4.6. The variable v in Fig. 4.8 is called the *true anomaly* in orbit mechanics. The problem of determining satellite position from these data and equations is called the *Kepler problem*. The hardest part of the Kepler problem is determining true anomaly as a function of time, which is nonlinear with respect to time. This problem was eventually solved by introducing two intermediate "anomaly" variables.

The *eccentric anomaly* (E) is defined as a geometric function of true anomaly, as shown in Fig. 4.6. Eccentric anomaly E is defined by projecting the satellite position on the elliptical orbit out perpendicular to the semimajor axis a and onto the circumscribed circle. Eccentric anomaly is then defined as the central angle to this projection point on the circle, as shown in Fig. 4.6.

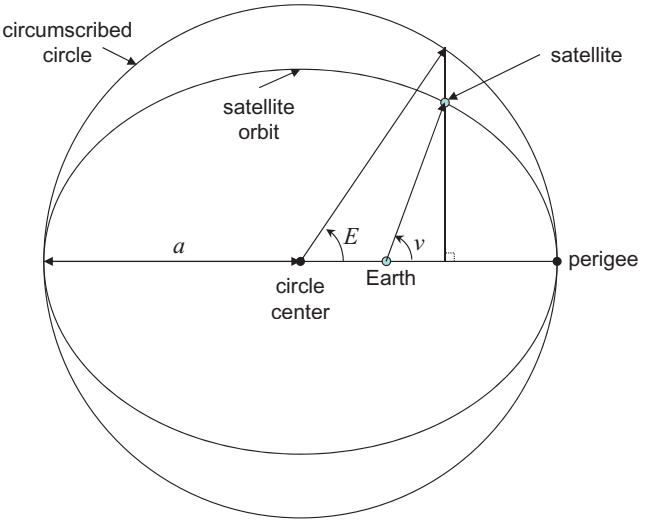


Fig. 4.6 Geometric relationship between true anomaly v and eccentric anomaly E.

The *mean anomaly* (M) can be described as the arc length (in radians) the satellite would have traveled since the perigee passage if it were moving in the circumscribed circle at the mean average angular velocity n. (Perigee is the point where the satellite is closest to the earth.) The mean anomaly can then be defined as shown in Eq. 4.3, which is a linear function of time:

$$M(t) = \frac{2\pi(t - t_{\text{perigee}})}{T_{\text{period}}}. \tag{4.3}$$

In Eq. 4.3, t is the time in seconds where the true anomaly is to be determined, t_{perigee} is the time when the satellite was at its perigee, and T_{period} is the period of the satellite orbit, in seconds to complete one full orbit. From this equation, calculating mean anomaly as a function of time is relatively easy, but the solution for true anomaly as a function of eccentric anomaly is much more difficult.

For the low eccentricities of GPS orbits, the numerical values of the true, eccentric, and mean anomalies are quite close together. However, the precision required in calculating true anomaly will require that they be treated as separate variables.

4.1.3.3 Kepler's Equation for the Eccentric Anomaly One equation in Table 4.2, shown below in Eq. 4.4, is called *Kepler's equation*. It relates the eccentric anomaly E_k for the satellite at the step time t_k to its mean anomaly, M_k, and the orbit eccentricity e. This equation is the most difficult of all the equations in Table 4.2 to solve for E_k as a function of M_k:

$$M_k = E_k - e\sin E_k. \tag{4.4}$$

The solution of Kepler's equation in Eq. 4.4 includes a transcendental function of eccentric anomaly E_k. It is impractical to solve for E_k in any way except by approximation. Standard practice is to solve the true anomaly equation iteratively for E_k, using the second-order Newton–Raphson method to solve for a residual term expressed in Eq. 4.5, and to make it arbitrarily small:

$$\varepsilon_k \equiv -M_k + E_k - e\sin E_k = 0. \tag{4.5}$$

Then, use the resulting value of E_k to calculate true anomaly; this process could continue at a given time step t_k until an arbitrarily small value is obtained. It starts by assigning an initial guess to E_k, say, $E_k[n]$ for $n = 0$. Since the GPS orbits are fairly circular to start with, the mean anomaly is a fair starting point. Then, form successively better estimates $E_k[n+1]$ by the second-order Newton–Raphson formula illustrated in Eq. 4.6:

$$E_k[n+1] = E_k[n] - \left\{ \frac{\varepsilon_k(E_k[n])}{\left.\frac{\partial \varepsilon_k}{E_k}\right|_{E_k = E_k[n]} - \left[\frac{\left.\frac{\partial^2 \varepsilon_k}{E_k^2}\right|_{E_k = E_k[n]} \varepsilon_k(E_k[n])}{2\left.\frac{\partial \varepsilon_k}{E_k}\right|_{E_k = E_k[n]}} \right]} \right\}. \tag{4.6}$$

The iteration of Eq. 4.6 can stop once the difference between the estimated $E_k[n+1]$ and the last value; that is, $E_k[n]$ is sufficiently small, for example, less than 10^{-6}. Usually, within about three to six iteration steps, at a particular t_k, the $E_k[n+1]$ value is within 10^{-6} of the $E_k[n]$ value, and iteration can stop.

4.1.3.4 Satellite Time Corrections The GPS Ground Control Segment estimates the satellite transmitter clock error, with respect to GPS time (t), and encodes clock bias, velocity, and acceleration error corrections, as well as a clock reference time in the broadcast navigation data stream. The user must correct the satellite clock error in accordance with Eq. 4.7:

$$t = t_{\mathrm{SV}i} - \Delta t_{\mathrm{SV}i}. \tag{4.7}$$

In Eq. 4.7, t is GPS system time in seconds, that is, corrected from, t_{SVi}, which is the effective SV PRN code phase time at message transmission time in seconds, and $\Delta t_{\mathrm{SV}i}$ is the total SV PRN code phase time offset in seconds. The SV PRN code phase offset (i.e., transmitter clock error) is given by Eq. 4.8.

The clock error is made up of a bias, velocity, acceleration, and a relativistic correction term:

$$\Delta t_{\mathrm{SV}i} = a_{fo} + a_{f1}(t - t_{\mathrm{oc}}) + a_{f2}(t - t_{\mathrm{oc}})^2 + \Delta t_{\mathrm{r}}, (\mathrm{s}) \tag{4.8}$$

where

a_{f0} = SV i clock bias error (s)
a_{f1} = SV i clock velocity error (s/s)
a_{f2} = SV i clock acceleration error ($\mathrm{s/s^2}$)
t_{oc} = clock reference time for clock correction date, (s)
Δt_{r} = SV i relativistic correction (s).

In Eq. 4.8, clock error polynomial coefficients, the reference time for the clock correction data, as well as information needed to calculate the relativistic correction, are provided in the broadcast navigation data.

The relativistic correction term, in seconds, is straightforward, as shown in Eq. 4.9, but needs the iterated eccentric anomaly E_k. In Eq. 4.9, the eccentricity and semimajor axis are as described in Table 4.2:

$$\Delta t_{\mathrm{r}} = \mathrm{F}e\sqrt{a}\sin E_k(\mathrm{s}) \tag{4.9}$$

where

$$F = \frac{-2\sqrt{\mu}}{c^2} = -4.442807622 \times 10^{-10}\ (\mathrm{s}/\sqrt{\mathrm{m}}).$$

Note that Eqs. 4.7 and 4.8 are coupled, but this will not cause problems. While the coefficients a_{f0}, a_{f1}, and a_{f2} are generated by using GPS time as indicated in Eq. 4.7, sensitivity of t, to the difference between $t_{\mathrm{SV}i}$ and t is negligible, when calculating the transmitter clock error expressed in Eq. 4.8. This negligible sensitivity will allow the user to approximate t by $t_{\mathrm{SV}i}$, using Eq. 4.8. The value of t must account for beginning- or end-of-week crossovers. Thus, if the quantity $t - t_{\mathrm{oc}}$ is greater than 302,400 s, subtract 604,800 s from t; if the quantity $t - t_{oc}$ is less than -302,400 s, then add 604,800 s to t.

The MATLAB® m-file (`ephemeris.m`) on the accompanying website calculates satellite positions for one set of ephemeris data and one time. Other programs calculate satellite positions for a range of time; see Appendix A.

4.1.4 C/A-Code and Its Properties

The C/A-code has the following functions:

1. *To Enable Accurate Range Measurements and Resistance to Errors Caused by Multipath.* To establish the position of a user to within reasonable

accurate (e.g., less than 100 m) satellite-to-user range estimates are needed. The estimates are made from measurements of signal propagation delay from the satellite to the user. To achieve the required accuracy in measuring signal delay, the GPS carrier must be modulated by a waveform having a relatively large bandwidth. The needed bandwidth is provided by the C/A-code modulation, which also permits the receiver to use correlation processing to effectively combat measurement errors. Because the C/A-code causes the bandwidth of the signal to be much greater than that needed to convey the 50-bps navigation data bit stream, the resulting signal is called a *spread-spectrum* signal. Using the C/A-code to increase the signal bandwidth also reduces errors in measuring signal delay caused by multipath (the arrival of the signal via multiple paths such as reflections from objects near the receiver antenna) since the ability to separate the direct path signal from the reflected signal improves as the signal bandwidth is made larger. While the C/A-code does a good job at this, the higher-rate spreading codes can do better at multipath mitigation.

2. *Permits Simultaneous Range Measurement from Several Satellites.* The use of a distinct C/A-code for each satellite permits all satellites to use the same frequencies without interfering with each other. This is possible because the signal from an individual satellite can be isolated by correlating it with a replica of its C/A-code in the receiver. This causes the C/A-code modulation from that satellite to be removed so that resulting signal bandwidth is narrowband; the signal part that remains is the low-rate navigation data. This process is called *despreading* of the signal. However, the correlation process does not cause the signals from other satellites to become narrowband because the codes from different satellites are nearly orthogonal. Therefore, the interfering signals from other satellites can be largely rejected by passing the desired despread signal through a narrowband filter, a bandwidth-sharing process called *code-division multiplexing* (CDM) or CDMA.
3. *Protection from Interference/Jamming.* The C/A-code also provides a measure of protection from intentional or unintentional interference or jamming of the received signal by another man-made signal. The correlation process that despreads the desired signal has the property of spreading other undesirable signals. Therefore, the signal power of any interfering signal, even if it is narrowband, will be spread over a large frequency band, and only that portion of the power lying in the narrowband filter will compete with the desired signal. The C/A-code provides about 20–30 dB of improvement in resistance to jamming from narrowband signals. Despite this distinct advantage, we must also remember that the GPS CDMA signals are relatively weak signals received on Earth.
4. *Short Period for Fast Acquisition.* The GPS C/A-code has a period of only 1 ms. This short period helps the GPS receiver to rapidly acquire

and track it. The short period and simplicity of its implementation also help reduce the cost, complexity, and power consumption in the GPS to support the C/A-code implementation.

We next detail important properties of the C/A-code.

4.1.4.1 Temporal Structure Each satellite has a unique C/A-code identified by the PRN number, but all the codes consist of a repeating sequence of 1023 chips occurring at a rate of 1.023 MHz with a period of 1 ms, as previously illustrated in Fig. 4.2. The leading edge of a specific chip in the sequence, called the *C/A-code epoch*, defines the beginning of a new period of the code. The code sequence can be generated digitally, with digital values, or converted into a bipolar signal with either positive or negative values without any loss of generality. The sequences of the individual 1023 chips appear to be random but are in fact generated by a deterministic algorithm implemented by a combination of shift registers (i.e., delay units) and combination logic. The algorithm produces two intermediate PRN code sequences (i.e., G1(t) and G2(t)). Then G2(t) is delayed with respect to G1(t), and they are combined to produce the C/A *Gold codes*. The details of the C/A-code generation can be found in Ref. 3. Gold codes are numerous to provide enough codes in the Gold code family to support the GPS constellation and other GNSS signal sources. While Gold codes do not have perfect correlation performance, their correlation properties are well understood and bounded [3]. Thus, the GPS C/A-code (based on the Gold codes) has the property of low cross correlation between different codes (nearly orthogonal) as well as reasonably small autocorrelation sidelobes.

4.1.4.2 Autocorrelation Function The autocorrelation of a code sequence refers to how well that code correlates to a delayed version of itself over time, theoretically over all time. An approximation of the autocorrelation function can be made over a limited observation time (T), for code $c(t)$ and a delayed version of that same code over a delay (τ), (i.e., $c(t-\tau)$) as shown in Eq. 4.10:

$$R_c(\tau) = \frac{1}{T}\int_0^T c(t)c(t-\tau)d\tau. \tag{4.10}$$

The autocorrelation of the GPS C/A can then be calculated from Eq. 4.10. An illustration of a GPS PRN code autocorrelation function (i.e., C/A-code) is shown in Fig. 4.7, where the width of one C/A-code chip is $t_c = 0.9775\,\mu s$. The shape of the code autocorelation is basically a triangle shape of width two chips at its base with a peak located at $\tau = 0$, when the codes are lined up in time. The autocorrelation function contains small sidelobes outside the triangular region. In most benign cases, these sidelobes do not have a major impact on ranging performance.

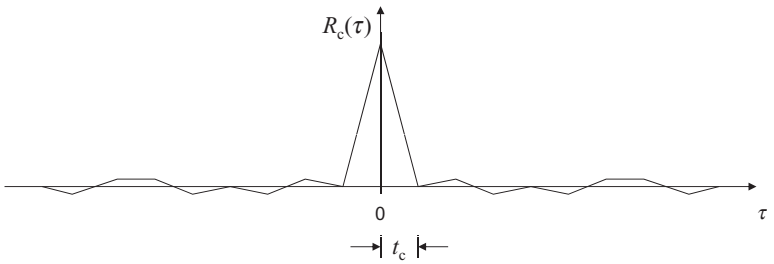


Fig. 4.7 Illustration of autocorrelation functions of GPS PRN codes.

The C/A-code autocorrelation function plays a substantial role in GPS receivers, inasmuch as it forms the basis for code tracking and accurate user-to-satellite range measurement estimation. In fact, the receiver continually computes values of this autocorrelation function in which the received signal code, $c(t)$, is correlated with a receiver-generated code (i.e., $c(t-\tau)$). Special hardware and software enable the receiver to adjust the reference waveform delay so that the value of τ is zero, thus enabling determination of the time of arrival of the received signal. As a final note, when the GPS signal is filtered in the GPS receiver, the autocorrelation function becomes rounded and accuracy is degraded.

4.1.4.3 Power Spectrum The power spectrum of a spreading code describes how the power in the code is distributed in the frequency domain. It can be defined in terms of either a Fourier series expansion of the code waveform or, equivalently, the code autocorrelation function. The power spectral density is the Fourier transform of the autocorrelation function. Using the code autocorrelation function $R_c(\tau)$, the power spectral density of our spreading code can be described a shown in Eq. 4.11:

$$S_c(f) = \lim_{T\to\infty} \frac{1}{T} \int_{-T}^{T} R_c(\tau) e^{-2\pi f\tau} d\tau. \tag{4.11}$$

A plot of $S_c(f)$ is shown as a smooth curve in Fig. 4.8. The overall envelope of the power spectral density of our spreading code is a sinc^2 function (i.e., $\sin^2(x)/x^2$ shape), with its nulls (i.e., zero values) at multiples of the code rate away from the central peak. Approximately 90% of the signal power is located between the first two nulls, but the smaller portion lying outside the first nulls is very important for accurate ranging. As for the GPS C/A-code, the spectrum nulls are at multiples of the C/A-code rate of 1.023 MHz. The period nature of the C/A-code with 1 ms code epochs will lead to spectral line components with 1-kHz spacing. Also shown in Fig. 4.8, for comparative purposes, is a typical noise power spectral density found in a GPS receiver after frequency conversion of the signal to baseband (i.e., with carrier removed). It can be seen

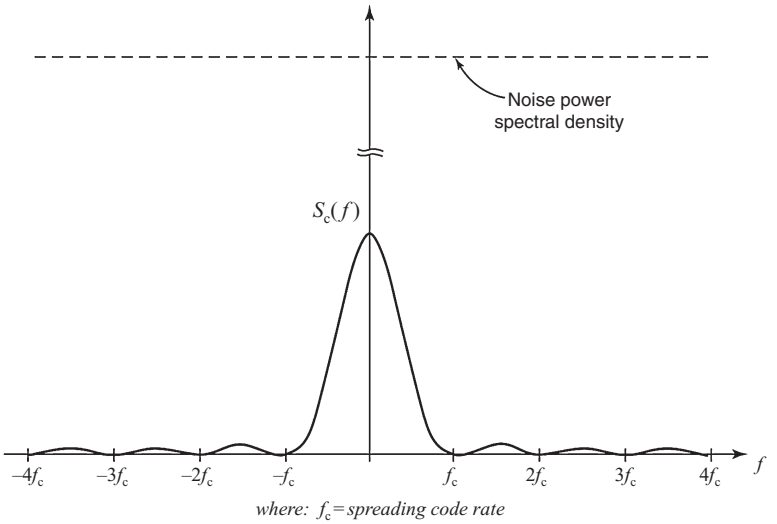


Fig. 4.8 Illustration of power spectrum of GPS spreading codes.

that the presence of the C/A-code causes the entire signal to lie well below the noise level because the signal power has been spread over a wide frequency range (approximately ±1 MHz).

4.1.4.4 Despreading of the Signal Spectrum Using the mathematical model of the signal modulated by the C/A-code presented in Eq. 4.1, within the GPS receiver, for a specific PRN code, the received signal's carrier frequency will be tracked with a phase-lock loop (PLL) to remove the carrier frequency, and only the data modulation $d(t)$ and the spreading code $c(t)$ modulation remain. For a particular PRN C/A-code, the resulting signal (at baseband [bb]), $s_{bb}(t)$, in normalized form is shown in Eq. 4.12:

$$s_{bb}(t) = d(t)c(t). \tag{4.12}$$

The power spectrum of the signal $s_{bb}(t)$ is similar to that of the C/A-code illustrated in Fig. 4.8. As previously mentioned, the signal in this form has a power spectrum lying below the receiver noise level. However, if this signal $s_{bb}(t)$ is multiplied by a replica of $c(t)$ in exact alignment with it (i.e., is punctual; we will call it $c_p(t)$), the resulting signal can then be fed through a signal recovery filter (e.g., low-pass filter [LPF]) function to produce an estimate of the original data, $\hat{d}(t)$, as represented in Eq. 4.13:

$$s_{bb}(t)c_p(t) = d(t)c(t)c_p(t) \xrightarrow[\text{LPF}]{} K\hat{d}(t). \tag{4.13}$$

As shown in Eq. 4.13, both of the coding sequences are considered to be of value ±1. This multiplication and low-pass filtering operation is a correlation, and the proportionality constant K has no consequential effect on our ability to produce a good estimate of the original navigation data bit stream.

This procedure is called *code despreading*, which removes the C/A-code modulation from the signal and exposes the underlying navigation data. The resulting signal has a two-sided spectral width of approximately 100 Hz due to the 50-bps navigation data modulation. From the above equation, it can be seen that the total signal power has not substantially changed in this process but is now contained in a much narrower bandwidth. Thus, the magnitude of the power spectrum is greatly increased, as indicated in Fig. 4.9. In fact, it now exceeds that of the noise, and the signal can be recovered by passing it through a small-bandwidth signal recovery filter to remove the wideband noise, as shown in Fig. 4.9.

4.1.4.5 Role of Despreading in Interference Suppression At the same time that the spectrum of the desired GPS signal is narrowed by the despreading process, any interfering signal that is not modulated by the same C/A-code will instead have its spectrum *spread* to a width of at least 2 MHz so that only a small portion of the interfering power can pass through the signal recovery filter. The amount of interference suppression gained by using the C/A-code depends on the bandwidth of the recovery filter, the bandwidth of

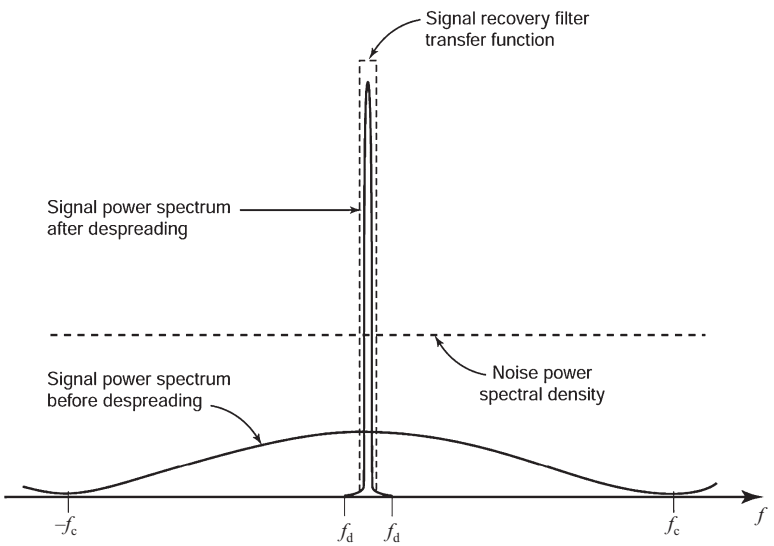


Fig. 4.9 Despreading of the spreading code.

the interfering signal, and the bandwidth of the C/A-code. For a narrowband interfering source whose signal can be modeled by a nearly sinusoidal waveform and a signal recovery filter bandwidth of 1000 Hz or more, the amount of interference suppression in decibels can be expressed in Eq. 4.14:

$$\eta = 10\log_{10}\left(\frac{W_c}{W_f}\right)(\text{dB}), \tag{4.14}$$

where

W_c = bandwidth of the spreading code
W_f = bandwidth of the filtering function.

In Eq. 4.14, the null-to-null bandwidth of the C/A-code is 2.046 MHz; and if $W_f = 2000$ Hz, about 30 dB of suppression can be obtained for a narrowband interference source. When the signal recovery filter has a bandwidth smaller than 1000 Hz, the situation is more complicated since the despread interfering sinusoid will have discrete spectral components with 1000-Hz spacing. As the bandwidth of the interfering signal increases, the C/A-code despreading process provides a decreasing amount of interference suppression. For interference sources having a bandwidth greater than that of the signal recovery filter, the amount of suppression in decibels provided by the spreading code can be approximated as shown in Eq. 4.15:

$$\eta = 10\log_{10}\left(\frac{W_I + W_c}{W_I}\right)(\text{dB}), \tag{4.15}$$

where

W_c = bandwidth of the spreading code
W_I = bandwidth of the interference source.

4.1.4.6 Cross-Correlation Function The cross correlation of a code sequence refers to how well that code correlates to a delayed version of a *different* code over time, theoretically over all time. An approximation of the cross-correlation function can be made over a limited observation time (T), for code $c_1(t)$ and a delayed version of a different code (i.e., $c_2(t-\tau)$) as shown in Eq. 4.16:

$$R_{c_1c_2}(\tau) = \frac{1}{T}\int_0^T c_1(t)c_2(t-\tau)d\tau. \tag{4.16}$$

Thus, when a selected satellite signal is encoded with a given spreading code, and it is despread using a replica of that same code, the signals from other satellites (encoded with different speading codes) look like wideband interference sources that are below the noise level. This permits a GPS receiver to extract a multiplicity of individual satellite signals and process them individually, even though all signals are transmitted at the same frequency. This is why this process is called CDM and allows for access to all of the GPS signals (i.e., CDMA).

4.1.5 P(Y)-Code and Its Properties

The GPS P(Y)-code is a higher-rate code, encrypted, and appears on both of the GPS L1 and L2 frequencies. It is used by military, authorized, and by civil users to produce "semicodeless" measurements. The P(Y)-code has the following functions:

1. *Increased Jamming Protection.* Because the bandwidth of the P-code is 10 times greater than that of the C/A-code, it offers approximately 10 dB more protection from narrowband interference. In military applications, the interference is likely to be a deliberate attempt to jam (render useless) the received GPS signal.
2. *Provision for Antispoofing.* In addition to jamming, another military tactic that an enemy can employ is to radiate a signal that appears to be a GPS signal (*spoofing*) but, in reality, is designed to confuse the GPS receiver. This is prevented by encrypting the P-code. The would-be spoofer cannot know the encryption process and cannot make the contending signal look like a properly encrypted signal. Thus, the receiver can reject the false signal and decrypt the desired one.
3. *Denial of P-Code Use.* The structure of the P-code is published in the open literature, so that anyone may generate it as a reference code for despreading the signal and for making range measurements. However, encryption of the P-code by the military will deny its use by unauthorized parties.
4. *Increased Code Range Measurement Accuracy.* All other parameters being equal, accuracy in range measurement improves as the signal bandwidth increases. Thus, the P-code provides improved range measurement accuracy as compared to the C/A-code. Simultaneous range measurements using both codes is even better. Because of its increased bandwidth, the P-code is also more resistant to range errors caused by multipath.

4.1.5.1 P-Code Characteristics Unlike the C/A-code, the P-code modulates both the L1 and L2 carriers. Its chipping rate is 10.23 MHz, which is

precisely 10 times the C/A rate, and it has a period of 1 week. It is transmitted synchronously with the C/A-code in the sense that each chip transition of the C/A-code always corresponds to a chip transition in the P-code. Like the C/A-code, the P-code autocorrelation function has a triangular central peak centered at $\tau = 0$, but with one-tenth the base width, as shown in Fig. 4.7, where the width of one P(Y)-code chip is $t_c = 0.09775\,\mu s$. The power spectrum also has a $\sin^2(x)/x^2$ characteristic, but with 10 times the bandwidth, as indicated in Fig. 4.8, where f_c is 10.23 MHz. Because the period of the P-code is so long, the power spectrum can be regarded as continuous envelope for practical purposes. Each satellite broadcasts a unique P-code. The technique used to generate it is similar to that of the C/A-code but somewhat more complicated. The details of the P-code generation can be found in Ref. 2.

4.1.5.2 Y-Code The encrypted form of the P-code used for antispoofing and denial of the P-code to unauthorized users is called the *Y-code*. The Y-code is formed by multiplying the P-code by an encrypting code called the *W-code*. The W-code is a random-looking sequence of chips that occur at a 511.5-kHz rate. Thus there are 20 P-code chips for every W-code chip. Since both the P-code and the W-code have chip values of ±1, the resulting P(Y)-code has the same appearance as the P-code; that is, it also has a 10.23-MHz chipping rate. However, the Y-code cannot be despread by a receiver replica P-code unless it is decrypted. Decryption consists of multiplying the Y-code by a receiver-generated replica of the W-code that is made available only to authorized users. Since the encrypting W-code is also not known by the creators of spoofing signals, it is easy to verify that such signals are not legitimate.

4.1.6 L1 and L2 Carriers

The L1 (or L2) carrier is used for the following purposes:

1. *Propagate the GPS signal from the satellite to the user in the service area provided by the system.*
2. *To provide very accurate relative range measurements for precision applications using carrier phase.*
3. *To provide accurate Doppler measurements.* The phase rate of the received carrier can be used for accurate determination of user velocity. The integrated Doppler, which can be obtained by counting the cycles of the received carrier, is often used as a precise delta range observable that can materially aid the performance of code tracking loops. The integrated Doppler history is also used as part of the carrier phase ambiguity resolution process.

4.1.6.1 Dual-Frequency Operation The use of *both* the L1 and L2 frequencies provides the following benefits:

1. *Provides Accurate Measurement of Ionosphere Signal Delay.* A major source of ranging error is caused by changes in both the phase velocity and group velocity of the signal as it passes through the ionosphere. Range errors of 10–20 m are commonplace and can be much larger (e.g., 100 m). Because the delay induced by the ionosphere is known to be inversely proportional to the square of frequency, ionosphere range error can be estimated accurately by comparing the times of arrival of the L1 and L2 signals. Details on the calculations appear in Chapter 7.
2. *Facilitates Carrier Phase Ambiguity Resolution.* In high-accuracy GPS differential positioning, the range estimates using carrier phase measurements are precise but highly ambiguous due to the periodic structure of the carrier. The ambiguity is more easily resolved (by various methods) as the carrier frequency decreases. By using L1 and L2 carrier frequencies, the ambiguity resolution can be based on their frequency difference (1575.42–1227.6 MHz), which is smaller than either carrier frequency alone, and hence will result in faster and more reliable ambiguity resolution performance.
3. *Provides System Redundancy (Primarily for the Military User).*

4.1.7 Transmitted Power Levels

The GPS signals are transmitted at a minimum power level on the order of 478-W (26.8-dBW) effective isotropic radiated power (EIRP), which means that the minimum received power is the same as would be obtained if the satellite radiated 478 W from an isotropic antenna. This effective power level is reached by radiating a smaller total power into a beam approximately 30° wide toward the earth. The radiated power level was chosen to provide a signal-to-noise ratio (SNR) sufficient for tracking of the signal by a receiver on the earth with an unobstructed view of the satellite and to not interfere with other terrestrial communication systems. The low signal power level provides a challenge to meet the need to operate GPS receivers under less desirable conditions, such as in heavy vegetation, in urban canyons, or indoor environments where considerable signal attenuation often occurs. These low signal power levels can also pose problems when attempting to operate a GPS receiver in a jamming or interference environment. The GPS P(Y) signal is transmitted at a power level that is 3 dB less on L1 and 6 dB less on L2 than the C/A-encoded signal on L1 as illustrated in Fig. 4.1.

4.1.8 Free Space and Other Loss Factors

As the signal propagates toward the earth, it loses power density due to spherical spreading. The loss is accounted for by a quantity called the *free-space loss factor* (FSLF), given in Eq. 4.17:

$$\text{FSLF} = \left(\frac{\lambda}{4\pi R}\right)^2. \tag{4.17}$$

With the height of a GPS satellite in a medium earth orbit (MEO) of approximately 20,000 km, and an L1 wavelength of 0.19 m, the FSLF is approximately 5.7×10^{-19} or –182.4 dB.

Other loss factors include an atmospheric loss factor (ALF) of 2.0 dB, which is allocated to account for any signal attenuation by the atmosphere. Additional provisions can be allocated for any antenna mismatch losses on the order of 2.4 dB.

A typical GPS antenna with right-hand circular polarization and a hemispherical pattern has about 3.0 dBi of gain relative to an isotropic antenna at zenith and will decrease as the elevation angle to the satellite decreases. The actual GPS signal power levels received are specified for a 3-dBi gain linearly polarized antenna with optimum orientation. With this linearly polarized antenna optimally placed with respect to the orientation of the incoming GPS signal, this antenna will only receive one-half of the power from the signal, so receiver antenna gain (G_r) of 0 dBic (dBs relative to a right-handed circular polarized signal) can be used. Additionally, there may be antenna orientation and mismatch losses that can be grouped and allocated.

4.1.9 Received Signal Power

When the EIRP is used, all of the other loss factors are considered, and applying the Friis transmission equation, the power received at the user antenna output can be calculated in dB format as EIRP + G_r + FSLF – ALF – Mismatch = 26.8 + 0 + (–182.4) – 2.0 – 2.4 = –156.0 dBW. Now the actual GPS signal power levels vary somewhat from this calculated value based on transmitter and user received antenna pattern variations, frequency, code type, and user latitude as documented in Ref. 2.

4.2 MODERNIZATION OF GPS

Since GPS was declared with a Final Operational Capability (FOC) in April 1995, applications and the use of GPS have evolved rapidly, especially in the civil sector. As a result, radically improved levels of performance have been reached in positioning, navigation, and time transfer. However, the availability of GPS has also spawned new and demanding applications that reveal certain shortcomings of the legacy system. Therefore, since the mid-1990s, numerous governmental and civilian committees have investigated these shortcomings and requirements for a more modernized GPS.

The modernization of GPS is a difficult and complex task that requires tradeoffs in many areas. Major issues include spectrum needs and availability, military and civil performance, signal integrity and availability, financing and

cost containment, and potential competing interests by other GNSSs and developing countries. Major decisions have been made for the incorporation of new civil frequencies and new civil and military signals that will enhance the overall performance of the GPS.

4.2.1 Areas to Benefit from Modernization

The areas that could benefit from a modernized GPS are the following:

1. *Robust Dual-Frequency Ionosphere Correction Capability for Civil Users.* Since only the encrypted P(Y)-code appears on the L2 frequency, civil users have lacked a robust dual-frequency ionosphere error correction capability. Civil users had to rely on semicodeless tracking of the GPS L2 signal, which is not as robust as access to a full-strength unencrypted signal. While civil users could employ a differential technique, this adds complexity to an ionosphere free user solution.
2. *A Better Civil Code.* While the GPS C/A-code is a good, simple spreading code, a better civil code would provide better correlation performance. Rather than just turning on the C/A-code on the L2 frequency, a more advanced spreading code would provide robust ranging and ionosphere error correction capability.
3. *Ability to Resolve Ambiguities in Phase Measurements Needs Improvement.* High-accuracy differential positioning at the centimeter level by civil users requires rapid and reliable resolution of ambiguities in phase measurements. Ambiguity resolution with single-frequency (L1) receivers generally requires a sufficient length of time for the satellite geometry to change significantly. Performance is improved with robust dual-frequency receivers. However, the effective SNR of the legacy P(Y) encrypted signal is dramatically reduced because the encrypted P(Y)-code cannot be despread by the civil user.
4. *Dual-Frequency Navigation Signals in the Aeronautical Radio Navigation Service (ARNS) Band.* The ARNS band of frequencies is federally protected and can be used for safety of life (SOL) applications. The GPS L1 band is in an ARNS band, but the GPS L2 band is not. (The L2 band is in the radio navigation satellite service [RNSS] band that has a substantial amount of uncontrolled signals in it.) In applications involving public safety, the integrity of the current system can be improved with a robust dual-frequency capability where both GPS signals are within the ARNS bands. This is particularly true in aviation landing systems that demand the presence of an adequate number of high-integrity satellite signals and functional cross-checks during precision approaches.
5. *Improvement Is Needed in Multipath Mitigation Capability.* Multipath remains a dominant source of GPS positioning error and cannot be removed by differential techniques. Although certain mitigation tech-

niques, such as multipath mitigation technology (MMT), approach theoretical performance limits for in-receiver processing, the required processing adds to receiver costs. In contrast, effective multipath rejection could be made available to all receivers by using new GPS signal designs.

6. *Military Requirements in a Jamming Environment.* The feature of selective availability (SA) was suspended at 8 p.m. EDT on May 1, 2000. SA was the degradation in the autonomous positioning performance of GPS, which was a concern in many civil applications requiring the full accuracy of which GPS is capable. If the GPS C/A-code was ever interfered with, the accuracies that it could be afforded would not be present. Because the P(Y)-code has an extremely long period (7 days), it is difficult to acquire unless some knowledge of the code timing is known. P(Y) timing information is supplied by the GPS HOW at the beginning of every subframe. However, to read the HOW, the C/A-code must first be acquired to gain access to the navigation message. Unfortunately, the C/A-code is relatively susceptible to jamming, which would seriously impair the ability of a military receiver to acquire the P(Y)-code. Direct P(Y) acquisition techniques are possible, but these techniques still require information about the satellite position, user position, and clock errors to be successful. Furthermore interference on the C/A-coded signal would also affect the P(Y)-coded signal in the same frequency band. It would be far better if direct acquisition of a high-performance code were possible, without the need to first acquire the C/A-code.

   Additional power received from the satellite would also help military users operate more effectively. While some advantages can be gained in the user equipment, having an increased power from the satellite could provide added value.

7. *Compatibility and Operability with Other GNSSs.* With the advances in other GNSSs by other nations, the requirement exists for international cooperation in the development of new GNSS signals to ensure they do not interfere with each other and potentially provide an interoperable combined GNSS service.

4.2.2 Elements of the Modernized GPS

Figure 4.10 illustrates the modernized GPS signal spectrum. The legacy GPS signals (L1 C/A and P(Y) and L2 P(Y)) as well as the additional modernized signals (L2C, L5, GPS L1 and L2 M-code, and L1C) are illustrated in Fig. 4.10.

The bandwidth available for the modernized GPS signals is up to 24 MHz, but the compatibility and power levels relative to the other codes are design considerations. Furthermore, assuming equal received power and filtered bandwidth, the ranging performance (with or without multipath) on a GPS signal is highly dependent upon the signal's spectral shape (or equivalently,

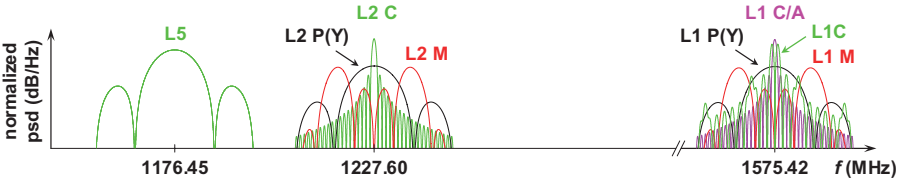


Fig. 4.10 A modernized GPS signal spectrum.

the shape of the autocorrelation function). In this sense, the L1 C/A-coded and L2 civil signals are somewhat equivalent in scope as are the P(Y) and L5 civil signals (albeit with very different characteristics). As we will see, the military M-coded signal and other GNSS codes are different because they use different subcarrier frequencies and chipping rates. These different subcarriers, in essence, add an aspect known as frequency division multiplexing to the GPS spectrum.

The major elements of these modernized signals are as follows.

4.2.3 L2 Civil Signal (L2C)

This new civil signal has a new code structure that has some performance advantages over the legacy C/A-code. The L2C signal offers civilian users the following improvements:

(a) *Robust Dual-Frequency Ionosphere Error Correction.* The dispersive delay characteristic of the ionosphere proportional to $1/f^2$ can be estimated much more accurately with this new, full-strength signal on the L2 frequency. Thus, civil users can choose to use a semicodeless P(Y) L2 and C/A L1, or a new L2C and C/A L1 technique to estimate the ionosphere delay.

(b) *Carrier Phase Ambiguity Resolution Will Be Significantly Improved.* The accessibility of the full-strength L1 and L2 signals provides "wide-lane" measurement combinations having ambiguities that are much easier to resolve.

(c) *The Additional L2C Signal Will Improve Robustness in Acquisition and Tracking.* The new spreading code identified for civil (C) users will provide more robust acquisition and tracking performance.

Originally, modernization efforts considered turning on the C/A-code at the L2 carrier frequency (1227.60 MHz) to provide the civilian community with a robust ionosphere correction capability as well as additional flexibility and robustness. However, later in the planning process, it was realized that additional advantages could be obtained by replacing the planned L2 C/A signal with a new L2 civil signal (L2C). The decision was made to use this new signal, and its structure was made public early in 2001. Both the L2C and the new

military M-code signal (to be described) appear on the L2 carrier orthogonal to the current P(Y).

Like the C/A-code, the C-code is a PRN code that runs at a 1.023×10^6 cps (chips per second) rate. However, it is generated by 2:1 time-division multiplexing of two independent subcodes, each having half the chipping rate, namely, 511.5×10^3 cps. Each of these subcodes is made available to the receiver by demultiplexing. These two subcodes have different periods before they repeat. The first subcode, the code moderate (CM), has a moderate length of 10,230 chips, a 20-ms period. The moderate length of this code permits relatively easy acquisition of the signal although the 2:1 multiplexing results in a 3-dB acquisition and data demodulation loss. The second subcode, the code long (CL), has a length of 707,250 chips, a 1.5-s period, and is data-free. The CM and CL codes are combined to provide the C-code at the 1.023-Mcps rate. Navigation data can be modulated on the C-code. Provisions call for no data, legacy navigation data at a 50-bps rate, or new civil navigation (CNAV) data at a 25 bps; the CNAV data at a 25-bps rate would be encoded using a rate one-half convolutional encoding technique, to produce a 50 sps (symbols per second) data bit stream that could then be modulated onto the L2 frequency to form the L2C signal. With no data, the coherent processing time can be increased substantially, thereby permitting better code and carrier tracking performance, especially at low SNR. The relatively long CL code length also generates smaller correlation sidelobes as compared to the C/A-code. Details on the L2 civil signal are given by Fontana et al. [4] and are provided in Ref. 2.

The existing C/A-code at the L1 frequency will be retained for legacy purposes.

4.2.4 L5 Signal

Although the use of the L1 and L2C signals can satisfy most civil users, there are concerns that the L2 frequency band may be subject to unacceptable levels of interference for applications involving public safety, such as aviation. The potential for interference arises because the International Telecommunications Union (ITU) has authorized the L2 band on a coprimary basis with radiolocation services, such as high-power radars. As a result of Federal Aviation Administration (FAA) requests, the Department of Transportation and Department of Defense have called for a new civil GPS frequency, called L5, at 1176.45 MHz in the ARNS band of 960–1215 MHz. To gain maximum performance, the L5 spread-spectrum codes were selected to have a higher chipping rate and longer period than do the C/A-codes to allow for better accuracy measurements. Additionally, the L5 signal has two signal components in phase quadrature, one of which will not carry data modulation. The L5 signal will provide the following system improvements:

(a) *Ranging Accuracy Will Improve.* Pseudorange errors due to random noise will be reduced below levels obtainable with the C/A-codes due

to the larger bandwidth of the proposed codes. As a consequence, both code-based positioning accuracy and phase ambiguity resolution performance will improve.

(b) *Errors due to Multipath Will Be Reduced.* The larger bandwidth of the new codes will sharpen the peak of the code autocorrelation function, thereby reducing the shift in the peak due to multipath signal components. The eventual multipath mitigation will depend upon the final receiver design and delay of the multipath.

(c) *Carrier Phase Tracking Will Improve.* Weak-signal phase tracking performance of GPS receivers is severely limited by the necessity of using a Costas (or equivalent-type) PLL to remove carrier phase reversals of the data modulation. Such loops rapidly degrade below a certain threshold (about 25–30 dB-Hz) because truly coherent integration of the carrier phase is limited to the 20-ms data bit length. In contrast, the "data-free" quadrature component of the L5 signal will permit coherent integration of the carrier for arbitrarily long periods, which will permit better phase tracking accuracy and lower tracking thresholds.

(d) *Weak-Signal Code Acquisition and Tracking Will Be Enhanced.* The data-free component of the L5 signal will also permit new levels of positioning capability with very weak signals. Acquisition will be improved because fully coherent integration times longer than 20 ms will be possible. Code tracking will also improve by virtue of better carrier phase tracking for the purpose of code rate aiding.

(e) *The L5 Signal Will Further Support Rapid and Reliable Carrier Phase Ambiguity Resolution.* The L5 signal is a full-strength, high-chipping rate code that will provide high-quality code and carrier phase measurements. These can be used to support various code and carrier combinations for high-accuracy carrier phase ambiguity resolution techniques.

(f) *The Codes Will Be Better Isolated from Each Other.* The longer length of the L5 codes will reduce the size of cross correlation between codes from different satellites, thus minimizing the probability of locking onto the wrong code during acquisition, even at the increased power levels of the modernized signals.

(g) *Advanced Navigation Messaging.* The L5 signal structure has a new CNAV messaging structure that will allow for increased data integrity.

GPS modernization for the L5 signal calls for a completely new civil signal format (i.e., L5 code) at a carrier frequency of 1176.45 MHz (i.e., L5 carrier). The L5 signal is defined in a quadrature scene where the total signal power is divided equally between in-phase (I) and quadrature (Q) components. Each component is modulated with a different but synchronized 10,230-chip direct sequence L5 code transmitted at 10.23 Mcps (the same rate as the P(Y)-code), but with a 1-ms period (the same as the C/A-code period). The I channel is modulated with a 100-sps data stream, which is obtained by applying rate 1/2,

constraint length 7, forward error correction (FEC) convolutional coding to a 50-bps navigation data message that contains a 24-bit cyclic redundancy check (CRC). The Q channel is unmodulated by navigation data. However, both channels are further modulated by Neuman–Hoffman (NH) synchronization codes, which provide additional spectral spreading of narrowband interference, improve bit and symbol synchronization, and also improve cross-correlation properties between signals from different GPS satellites. The L5 signal is shown in Fig. 4.10 illustrating the modernized GPS (and legacy GPS) signal spectrum.

Compared to the C/A-code, the 10-times larger chip count of the I and Q channel civil L5 codes provides lower autocorrelation sidelobes, and the 10 times higher chipping rate substantially improves ranging accuracy, provides better interference protection, and substantially reduces multipath errors at longer path separations (i.e., long delay multipath). Additionally, these codes were selected to reduce, as much as possible, the cross correlation between satellite signals. The absence of data modulation on the Q channel permits longer coherent processing intervals in code and carrier tracking loops, with full-cycle carrier tracking in the latter. As a result, the tracking capability and phase ambiguity resolution become more robust.

Further details on the civil L5 signal can be found in References 5–8.

4.2.5 M-Code

The new military (M) codes will also be transmitted on both the L1 and L2 carrier frequencies. These M-codes are based on a new family of split-spectrum GNSS codes for military and new GPS civil signals [9]. The M-codes will provide the following advantages to military users:

(a) *Direct Acquisition of the M-Codes Will Be Possible.* The design of these codes will eliminate the need to first acquire the L1 C/A-code with its relatively high vulnerability to jamming.

(b) *Better Ranging Accuracy Will Result.* As can be seen in Fig. 4.10, the M-codes have significantly more energy near the edges of the bands, with a relatively small amount of energy near the band center. Since most of the C/A-code power is near the band center, potential interference between the codes is mitigated. The effective bandwidth of the M-codes is much larger than that of the P(Y)-codes, which concentrate most of their power near the L1 or L2 carrier. Because of the modulated subcarrier, the autocorrelation function of the M-codes has not just one peak but several peaks spaced one subcarrier period apart, with the largest at the center. The modulated subcarrier will cause the central peak to be significantly sharpened, significantly reducing pseudorange measurement error.

(c) *Error due to Multipath Will Be Reduced.* The sharp central peak of the M-code autocorrelation function is less susceptible to shifting in the presence of multipath correlation function components.

The M-coded signals will be transmitted on the L1 and L2 carriers, with the capability of using different codes on the two frequencies. The M-codes are known as binary offset carrier (BOC) encoded signals where the notation of BOC(f_{sx}, f_{cx}) is used where f_{sx} represents the subcarrier multiplier, and f_{cx} represents the code rate multiplier, with respect to a nominal code rate of 1.023 MHz. The M-code is a BOC(10,5) code in which a 5.115-Mcps chipping sequence modulates a 10.23-MHz square wave subcarrier. Each spreading chip subtends exactly two cycles of the subcarrier, with the rising edge of the first subcarrier cycle coincident with initiation of the spreading chip. The spectrum of the BOC(10,5) code has considerably more relative power near the edges of the signal bandwidth than any of the C/A, P(Y), L2C, and L5 coded signals. As a consequence, the M-coded signal has minimal spectral overlap with the other GPS transmitted signals, which permits transmission at higher power levels without mutual interference. The resulting spectrum has two lobes, one on each side of the band center, thereby producing the split-spectrum code. The M-code signals are illustrated in Fig. 4.10. The M-code signal is transmitted in the same quadrature channel as the C/A-code (i.e., with the same carrier phase), that is, in phase quadrature with the P(Y)-code. The M-codes are encrypted and unavailable to unauthorized users. The nominal received power level is –158 dBW at Earth. Additional details on the BOC(10,5) code can be found in a paper by Barker et al. [10].

4.2.6 L1C Signal

A proposed new L1 Civil (L1C) signal is planned for the next generation of GPS SVs. Although the current C/A-code is planned to remain on the L1 frequency (1575.42 MHz), the additional L1C signal will add a higher-performance civil signal at the L1 frequency with potential interoperability with other GNSSs. Like the L5 civil signal, the planned L1C signal will have a data-free quadrature component.

The original L1C signal structure considered a pure BOC(1,1) signal but evolved into a more complex signal that multiplexes two BOC signals. Additional complexities involved the desire to have the L1C signal interoperable with other GNSS signals as well as potential intellectual property issues. The L1C signal contains a dataless (i.e., pilot) and data signal component transmitted in quadrature. The L1C signal for GPS that emerged from development is based upon a time-multiplexed BOC (TMBOC) modulation technique that synchronously time-multiplexes the BOC(1,1) and BOC(6.1) spreading codes for the pilot component (designated as $L1C_p$), and a BOC(1,1) modulated signal with navigation data (designated as $L1C_D$). For both, the BOC codes are generated synchronously at a rate of 1.023 MHz and are based on the Legendre sequence called Weil code. These codes have a period of 10 ms, so 10,230 chips are within one period. Additionally, there is an overlay code that is encoded onto the $L1C_p$ pilot channel. One bit of the overlay code has a duration and is synchronized to the 10-ms period of the BOC code generators. The overlay code rate is 100 bps and has 1800 bits in an 18-s period.

To generate the TMBOC signal for the $L1C_p$ channel, the BOC(1,1) and BOC(6,1) spreading sequences are time-multiplexed. With 33 symbols of a BOC(1,1) sequence, four symbols are replaced with BOC(6,1) chips. These occur at symbols 0, 4, 6, and 29. Thus, with a 75% of the power planned for distribution in the pilot signal, there will be 1/11th of the power in the BOC(6,1) component and 10/11th of the power in the BOC(1,1) component of the carrier.

The L1C signal also has a new navigation message structure, designated as CNAV-2, with three different subframe formats defined. Subframe 1 contains GPS time information (i.e., time of interval [TOI]). Subframe 2 contains ephemeris and clock correction data. Subframe 3 is commutated over various pages and provides less time-sensitive data such as almanac, UTC, and ionosphere that can be expended in the future.

The split-spectrum nature of the L1C encoded signal will provide some frequency isolation from the L1 C/A-encoded signal. Each spreading chip subtends exactly one cycle of the subcarrier, with the rising edge of the first subcarrier half-cycle coincident with initiation of the spreading chip. The TMBOC codes provide a larger root mean square (RMS) bandwidth compared to pure BOC(1,1).

For many years now, cooperation at the international level has been ongoing to enable the L1C signal to be interoperable with other GNSSs. A combined interoperable signal would allow a user to ubiquitously use navigation signals from different GNSSs with known and specified performances attributes.

Additional details on the L1C signal can be found in Refs. 11–13.

4.2.7 GPS Satellite Blocks

The families of satellites launched prior to recent modernization efforts are referred to as Block I (1978–1985), Block II (1989–1990), and Block IIA (1990–1997); all of these satellites transmit the legacy GPS signals (i.e., L1 C/A and P(Y) and L2 P(Y)). (The U.S. Naval Observatory has an up-to-date listing of all of the GPS satellites in use today [14].)

In 1997, a new family, the Block IIR satellites, began to replace the older Block II/IIA family. The Block IIR satellites have several improvements, including reprogrammable processors enabling problem fixes and upgrades in flight. Eight Block IIR satellites were modernized (designated as Block IIR-M) to include the new military M-code signals on both the L1 and L2 frequencies, as well as the new L2C signal on L2. The first Block IIR-M was launched in September 2005.

To help secure the L5 frequency utilization, one of the Block IIR-M satellites (GPS IIR-20(M)), SV49, was outfitted with a special L5 payload and was launched on March 24, 2009. This particular satellite had hardware configuration issues relating to the L5 payload installation and is transmitting a degraded signal. Since that time, the navigation signals have been set unhealthy in the broadcast navigation message.

The Block IIF (i.e., follow-on) family was the next generation of GPS satellites, retaining all the capabilities of the previous blocks, but with many improvements, including an extended design life of 12 years, faster processors with more memory, and the inclusion of the new L5 signal on a third, L5 frequency (1176.45 MHz). The first Block IIF satellite was launched in May 2010.

4.2.8 GPS III

The next block of GPS satellites planned is designated as the Block III family, which is still under development. The GPS III block of satellites and associated GPS Ground Operational Control Segment (OCX) components will represent a major advancement in capabilities for military and civil users. GPS III is planned to include all of the legacy and modernized GPS signal components, including the new L1C signal, and to add specified signal integrity. The added signal integrity planned for in GPS III may be able to satisfy some of the aviation requirements [15]. Improvements for military users include two high-power spot beams for the L1 and L2 military M-code signals, providing 20-dB higher received power over the earlier M-code signals. However, in the fully modernized Block III satellites, the M-coded signal components are planned to be radiated as physically distinct signals from a separate antenna on the same satellite. This is done in order to enable optional transmission of a spot beam for greater antijam resistance within a selected local region on the earth.

4.3 GLONASS SIGNAL STRUCTURE AND CHARACTERISTICS

GLONASS is the Russian GNSS. The GLONASS has similar operational requirements to GPS with some key differences in its configuration and signal structure. Like GPS, GLONASS is an all-weather, 24-h satellite-based navigation system that has space, control, and user segments. The first GLONASS satellite was launched in 1982, and the GLONASS was declared an operational system on September 24, 1993.

The GLONASS satellite constellation is designed to operate with 24 satellites in three orbital planes at 19,100-km altitude (whereas GPS uses six planes at 20,180-km altitude). GLONASS calls for eight SVs equally spaced in each plane. The GLONASS orbital period is 11 h 15 min, which is slightly shorter than the 11-h 56-min 02-sec orbital period for a GPS satellite. Because some areas of Russia are located at high latitudes, the orbital inclination of 64.8° is used as opposed to the inclination of 55° used for GPS. Each GLONASS satellite transmits its own ephemeris and system almanac data. Via the GLONASS Ground Control Segment, each GLONASS satellite transmits its position, velocity, and lunar/solar acceleration effects in an ECEF coordinate frame (as opposed to GPS, that encodes SV positions using Keplerian orbital

parameters). GLONASS ECEF coordinates are referenced to the PZ-90.02 datum and time reference linked to their national reference of UTC (SU) (Soviet Union, now Russia).

4.3.1 Frequency Division Multiple Access (FDMA) Signals

GLONASS uses multiple frequencies in the L-band and has used frequencies separated by a substantial distance for ionosphere mitigation (i.e., L1 and L2), but these are slightly different from the GPS L1 and L2 frequencies. One significant difference between GLONASS and GPS is that GLONASS has historically used an FDMA architecture as opposed to the CDMA approach used by GPS.

4.3.1.1 Carrier Components The GLONASS uses two L-band frequencies, L1 and L2, as defined in Eq. 4.18. The channel numbers for GLONASS signal operation are:

$$\begin{aligned} f_{K1} &= f_{01} + K\Delta f_1 \\ f_{K2} &= f_{02} + K\Delta f_2, \end{aligned} \tag{4.18}$$

where

K = channel number, $(-7 \le K \le +6)$
f_{01} = 1602 MHz; Δf_1 = 562.5 kHz
f_{02} = 1246 MHz; Δf_1 = 437.5 kHz.

4.3.1.2 Spreading Codes and Modulation With the GLONASS signals isolated in frequency, an optimum maximal-length (m-sequence) can be used as the spreading code. GLONASS utilizes two such codes, one standard precision navigation signal (SPNS) at a 0.511-Mcps rate that repeats every 1 ms and a second High-Precision Navigation Signal (HPNS) at a 5.11-Mcps rate that repeats every 1 s. Similar to GPS, the GLONASS signals utilize BPSK modulation and are transmitted out of a right-hand circularly polarized (RHCP) antenna.

4.3.1.3 Navigation Data Format The format of the GLONASS navigation data is similar to the GPS navigation data format, with different names and content. The GLONASS navigation data format is organized as a superframe that is made up of frames, where frames are made up of strings. A superframe has a duration of 150 s and is made up of five frames, so each frame lasts 30 s. Each frame is made up of 15 strings, where a string has a duration of 2 s. GLONASS encodes satellite ephemeris data as immediate data and almanac data as nonimmediate data. There is a time mark in the GLONASS navigation data (last 0.3 s of a string) that is an encode PRN sequence.

4.3.1.4 Satellite Families While the first series of GLONASS satellites were launched from 1982 to 2003, the GLONASS-M satellites were launched beginning in 2003. These GLONASS-M satellites had improved frequency plans and the accessible signals that are known today. The new generation of GLONASS satellites is designated as GLONASS-K satellites and are considered a major modernization effort by the Russian government. These GLONASS-K satellites plan to transmit the legacy GLONASS FDMA signals as well as a new CDMA format.

Additional details of the GLONASS signal structure and GLONASS-M satellite capabilities can be found in Ref. 16.

4.3.2 CDMA Modernization

One of the issues with an FDMA GNSS structure is the interchannel (i.e., interfrequency) biases that can arise within the FDMA GNSS receiver. If not properly addressed in the receiver design, these interchannel biases can be a significant error source in the user solutions. These error sources arise because the various navigation signals pass through the components within the receiver at slightly different frequencies. The group delay thought these components are noncommon, at the different frequencies, and produced different delays on the various navigation signals, coming from different satellites. This interfrequency bias is substantially reduced (on a comparative basis) with CDMA-based navigation systems because all of the signals are transmitted at the same frequency. (The relatively small amount of Doppler received from the various CDMA navigation signals is minor when considering the group delay.)

An additional consideration with an FDMA GNSS signal structure is the amount of frequency bandwidth that is required to support the FDMA architecture. CDMA architecture typically has all the signals transmitted at the same carrier frequency, for more efficient utilization of a given bandwidth.

GLONASS has established several separate versions of its GLONASS-K satellites. The first GLONASS-K1 satellite was launched on February 26, 2011, which carried the first GLONASS CDMA signal structure and has been successfully tracked on Earth [17]. The GLONASS-K1 satellite has transmitted a CDMA signal at a designated L3 frequency of 1202.025 MHz (test signal), as well as the legacy GLONASS FDMA signals at L1 and L2. The CDMA signal from the GLONASS-K1 satellite is considered a test signal. The follow-on generation of satellites is designated as the GLONASS-K2 satellites [18]. A full constellation of legacy and new CDMA signals are planned for these GLONASS-K2 satellites including plans to transmit its CDMA signal on or near the GPS L1 and L2 frequency bands. The GLONASS KM satellites are in the research phase, and plans call for the transmission of the legacy GLONASS FDMA signals, the CDMA signals introduced in the GLONASS-K2, and a new CDMA signal on the GPS L5 frequency.

4.4 GALILEO

Galileo is a GNSS being developed by the European Union and the European Space Agency (ESA). Like GPS and GLONASS, it is an all-weather, 24-h satellite-based navigation system being designed to provide various services. The program has had three development phases: (1) definition (completed), (2) development and launch of on-orbit validation satellites, and (3) launch of operational satellites, including additional development [19]. The first Galileo In-Orbit Validation Element (GIOVE) satellite, designated as GIOVE-A, was launched in December 28, 1995, followed by the GIOVE-B on April 27, 2008. The next two Galileo operational satellites were launched on October 21, 2011 to provide additional validation of the Galileo.

4.4.1 Constellation and Levels of Services

The full constellation of Galileo is planned to have 30 satellites in MEOs with an orbital radius of 23,222 km (similar to the GPS orbital radius of 20,180 km). The inclination angle of the orbital plane is 56° (GPS is 55°), with three orbital planes (GPS has six). This constellation will thus have 10 satellites in each orbital plane.

Various services are planned for Galileo, including Open Service (OS), Commercial Service (CS), Public Regulated Services (PRS), a SOL, and Search and Rescue (SAR) service. These services will be supported with different signal structures and encoding formats tailored to support the particular service.

4.4.2 Navigation Data and Signals

Table 4.3 lists some of the key parameters for the Galileo that will be discussed in this section. The table lists the Galileo signals, frequencies, identifiable signal

TABLE 4.3. Key Galileo Signals and Parameters

Signal	Frequency (MHz)	Component	Data	Service
E1	**1575.420**			
		E1-B	I/NAV	OS/CS/SOL
		E1-C	Pilot	
E6	**1278.750**			
		E6-B	C/NAV	CS
		E6-C	Pilot	
E5	**1191.795**			
E5a	1176.450	E5a-I	F/NAV	OS
		E5a-Q	Pilot	
E5b	1207.140	E5b-I	I/NAV	OS/CS/SOL
		E5b-Q	Pilot	

component, its navigation data format, and what service the signal is intended to support. The European Union has published the Galileo OS Interface Control Document, which contains significant detail on the Galileo signals in space [20]. All of the Galileo signals transmit two orthogonal signal components, where the in-phase component transmits the navigation data and the quadrature component is dataless (i.e., a pilot). These two components have a power sharing so that the dataless channel can be used to aid the receiver in the acquisition and tracking of the signal. All of the individual Galileo GNSS signal components utilize phase-shift keying modulation and RHCP for the navigation signals.

To support the various services for Galileo, three different navigation formats are planned for implementation: (1) a free navigation (F/NAV) format to support OS in for the E1 and E5a signals; (2) the integrity navigation (I/NAV) format is planned to support SOL services for the L1 and E5b signals; and (3) a commercial navigation (C/NAV) format is to support CS.

Galileo E1 frequency (same as GPS L1) at 1574.42 MHz will have a split-spectrum-type signal around the center frequency. This signal is planned to interoperate with the GPS L1C signal; however, discussions continue on the technical, political, and intellectual property aspects with its implementation. Despite these challenges, the Galileo E1 signal is planned to be a combined BOC (CBOC) signal that is based upon two BOC signals (a BOC(1,1) and BOC(6.1) basis component) in phase quadrature. The E1 in-phase component has I/NAV data encoded on it and the quadrature phase has no data (i.e., pilot).

The Galileo E6 signal is planned to support CS at a center frequency of 1278.750 MHz, with no offset carrier, and a spreading code at a rate of 5.115 MHz (5×1.023 MHz). The E6 signal has two signal components in quadrature, where the C/NAV message format is encoded on the in-phase component, and no data are on the quadrature component.

The E5 signal is a unique GNSS signal that has an overall center frequency of 1191.795 MHz, with two areas of maximum power at 1176.450 and 1207.140 MHz. The wideband E5 signal is generated by a modulation technique called alternate binary offset carrier (altBOC). The generation of the E5 signal is such that it is composed of two Galileo signals that can be received and processed separately or combined by the user. The first of these two signals within the composite E5 signal is the E5a, centered at 1176.450 MHz (same as the GPS L5). The E5a signal has again two signal components, transmitted in quadrature, where one has data (in-phase) and other does not (quadrature). The F/NAV data are encoded onto the in-phase E5a channel at a 50-sps rate. The second of these two signals within the composite E5 signal is the E5b, centered at 1207.140 MHz. The E5b signal has two signal components, transmitted in quadrature, where one has data (in-phase) and the other does not (quadrature). The in-phase channel has the I/NAV message format at a 250-sps rate to support OS, CS, and SOL applications. The data encoded within the I/NAV format will contain important integrity information necessary to support SOL applications [21].

4.5 COMPASS/BD

Compass is the Chinese developed GNSS, where Compass is an English translation for BD. The Chinese government performed initial development on the Compass in the 2000–2003 timeframe with reference to BD-1. Current Compass development is often referred to as BD-2. The Compass space segment is called a Dipper constellation that plans for an eventual 27 MEO, 5 geostationary earth orbits (GEOs), and 3 inclined GEOs (IGSO) satellites. The GEOs are planned for longitude locations at: 58.75°E, 80°E, 110.5°E, 140°E, and 160°E. The system will use its own datum, China Geodetic Coordinate System 2000 (CGCS2000), and time reference BeiDou time (BDT) system, relatable to UTC. Of the BD-2 SVs, the first MEO Compass M-1 satellite was launched on April 14, 2007. The GEOs have been launched with an orbital radius of 42,164.17 km. The first IGSO SV was launched on July 31, 2010 with an inclination angle of 55° (same as GPS MEOs).

While there are multiple global and regional services planned for Compass, only a test version of the interface control document (ICD) has been released [21]. The ICD does identify the three development phases of the Compass/BD and the system frequency and signal characteristics are expected to change in the migration from the current Phase II to Phase III. The current ICD specifies a B1 GNSS signal at a center frequency of 1561.098 MHz. The architecture is based on a CDMA approach. The spreading codes used are Gold codes, based on 11 stages of delay, running at 2.046 Mcps, so the code will repeat after 1 ms. Navigation data on the in-phase channel of the MEO and IGSO SVs is at a rate of 50 bps, with a secondary code rate of 1 kbps. Navigation data on the in-phase channel for the GEOs is at a 500-bps rate.

4.6 QZSS

The QZSS Navigation Service is a regional space-based positioning system being developed by the Japan Aerospace Exploration Agency (JAXA). The QZSS is to be interoperable with GPS and provides augmentation to the GPS over the regions covered by a particular Quasi-Zenith Satellite (QZS). The QZSS also includes additional services and signals to support a variety of users. Details of the QZSS and how to interface to the signal in space can be found in Ref. 22.

Table 4.4 illustrates the various navigation signals for the QZSS. Some of the QZSS signals are similar to the GPS signals (L1 C/A, L2C, L5, and L1C); the L1-submeter accuracy with integrity function (SAIF) signal is a space-based augmentation system (SBAS)-type signal, and the LEX is an experimental signal.

The QZSs are in a highly inclined elliptical orbit (HEO), that is, geosynchronous with the earth rotation rate. There are provisions for three QZSs

TABLE 4.4. Key QZSS Signals and Parameters

Signal	Frequency (MHz)	Bandwidth (MHz)	Comment
L1C/A L1C	1575.42	24	GPS interoperable signal (current and future)
L2C	1227.60	24	
L5	1176.45	24.9	
L1-SAIF	1575.42	24	Compatible with GPS-SBAS WDGPS
LEX	1278.75	39	Experimental signal with high data rate (2 kbps) (same carrier as Galileo E6 signal)

with their semimajor axis of 42,164 km at an inclination of 43°. There are plans to place QZSs at longitude locations corresponding to right ascension of ascending node (RAAN) of 90°, 210°, and 330°. The first phase of the QZSS produced the launch of the first Michibiki QZS on September 11, 2010 and is centered at a longitude of approximately 135°E. (Michibiki means to guide or lead the way.) This satellite is in operation at an approximate location of a RAAN equal to 195° (i.e., longitudually centered over Japan). Two additional QZSs are planned. For a user in the Southeast Asia area, the QZS will track out a "figure eight" in the sky. Most users in that area will have good visibility to the QZS 24 h a day with a slowly varying Doppler frequency.

The L1 C/A, L2C, L5, and L1C codes are similar to the GPS codes and use different Gold code PRNs identified in the QZSS IS [23]. The L1-SAIF signal is an additional ranging signal and provides the augmentation data to GPS. The L1-SAIF signal has a data rate of 250 bps with a rate one-half convolution code to produce a 500 sps and is a wide area differential GPS (DGPS) that is a GPS-SBAS, and implements the specification defined by the Radio Technical Commission for Aeronautics (RTCA) [23].

The LEX signal is considered an L-band experimental signal and implements a spreading code known as a Kasami sequence. The LEX signal is formed by chip-by-chip multiplexing of two of the sequences. Data are encoded onto the small Kasami sequence at a 250-sps rate for a net data rate of 2 kbps. A block Reed–Solomon code is added for FEC. The quadrature phase of the LEX is modulated with the long Kasami sequence with no navigation data. The LEX spreading code runs at a rate of 5.115 MHz (5×1.023 MHz), using BPSK, so it is often shown as BPSK(5).

PROBLEMS

4.1 An important signal parameter is the maximum Doppler shift due to satellite motion, which must be accommodated by a receiver. Find its approximate value by assuming that a GPS satellite has a circular orbit with a radius of 27,000 km, an inclination angle of 55°, and a 12-h period. Is the rotation rate of the earth significant? At what latitude(s) would one expect to see the largest possible Doppler shift?

4.2 Another important parameter is the maximum *rate* of Doppler shift in hertz per second that a PLL must be able to track. Using the orbital parameters of the previous problem, calculate the maximum rate of Doppler shift of a GPS signal one would expect, assuming that the receiver is stationary with respect to the earth.

4.3 Find the power spectrum of the 50-bps data stream containing the navigation message. Assume that the bit values are –1 and 1 with equal probability of occurrence, that the bits are uncorrelated random variables, and that the location of the bit boundary closest to $t = 0$ is a uniformly distributed random variable on the interval (–0.01 s, 0.01 s). (*Hint*: First find the autocorrelation function $R(\tau)$ of the bit stream and then take its Fourier transform.)

4.4 In two-dimensional positioning, the user's altitude is known, so only three satellites are needed. Thus, there are three pseudorange equations containing two position coordinates (e.g., latitude and longitude) and the receiver clock bias term B. Since the equations are nonlinear, there will generally be more than one position solution, and all solutions will be at the same altitude. Determine a procedure that isolates the correct solution.

4.5 Some civil receivers attempt to extract the L2 carrier by squaring the received waveform after it has been frequency-shifted to a lower IF. Show that the squaring process removes the P(Y)-code and the data modulation, leaving a sinusoidal signal component at twice the frequency of the original IF carrier. If the SNR in a 20-MHz IF bandwidth is –30 dB before squaring, find the SNR of the double-frequency component after squaring if it is passed through a 20-MHz bandpass filter. How narrow would the bandpass filter have to be to increase the SNR to 0 dB?

4.6 The relativistic effect in a GPS satellite clock, which is compensated by a deliberate clock offset, is about

(a) 4.5 parts per million

(b) 4.5 parts per 100 million

(c) 4.5 parts per 10 billion

(d) 4.5 parts per trillion

4.7 The following component of the ephemeris error contributes the most to the range error:

(a) along-track error

(b) cross-track error

(c) both along-track and cross-track errors

(d) radial error.

4.8 The differences between pseudorange and carrier phase observations are

(a) integer ambiguity, multipath errors, and receiver noise

(b) satellite clock, integer ambiguity, multipath errors, and receiver noise

(c) integer ambiguity, ionosphere errors, multipath errors, and receiver noise

(d) satellite clock, integer ambiguity, ionosphere errors, multipath errors, and receiver noise.

4.9 GPS WN started incrementing from zero at

(a) midnight of January 5–6, 1980

(b) midnight of. January 5–6, 1995

(c) midnight of December 31–January 1, 1994–1995

(d) midnight of December 31–January 1, 1999–2000.

4.10 The complete set of GPS satellite ephemeris data comes once in every

(a) 6 s

(b) 18 s

(c) 30 s

(d) 150 s.

4.11 Describe how the time of travel (from satellite to receiver) of the GPS signal is determined.

4.12 Calculate the time of the next GPS week rollover event.

4.13 How far does a Galileo satellite move during the time it takes the signal to leave the satellite and be received at Earth? Use a nominal transit time in your calculation.

REFERENCES

[1] DOD, *Global Positioning System Standard Positioning Service Performance Standard*, 4th ed. DOD ASD(NII)/DASD, Washington, DC, 2008.

[2] GPS Directorate, Systems Engineering & Integration Interface Specification, Navstar GPS Space Segment/Navigation User Segment Interfaces, IS-GPS-200, IS-GPS-200F, 21-Sept-2011, available at: http://www.navcen.uscg.gov/pdf/IS-GPS-200F.pdf, visited July 14, 2012.

[3] R. Gold, "Optimal Binary Sequences for Spread Spectrum Multiplexing," *IEEE Transactions on Information Theory* **13**(4), 619–621 (1967).

[4] R. D. Fontana, W. Cheung, P. M. Novak, and T. A. Stansell, "The New L2 Civil Signal," *Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001)*, Salt Lake City, UT, September 2001, pp. 617–631.

[5] GPS Directorate, Systems Engineering & Integration Interface Specification, IS-GPS-705, Navstar GPS Space Segment/User Segment L5 Interface, IS-GPS-705B, 21-Sept-2011, available at: http://www.navcen.uscg.gov/pdf/gps/IS-GPS-705B.pdf, visited July 14, 2012.

[6] A. J. Van Dierendonck and J. J. Spilker, Jr., "Proposed Civil GPS Signal at 1176.45 MHz: In-Phase/Quadrature Codes at 10.23 MHz Chip Rate," *Proceedings of the 55th Annual Meeting of The Institute of Navigation*, Cambridge, MA, June 1999, pp. 761–770.

[7] C. Hegarty and A. J. Van Dierendonck, "Civil GPS/WAAS Signal Design and Interference Environment at 1176.45 MHz: Results of RTCA SC159 WG1 Activities," *Proceedings of the 12th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1999)*, Nashville, TN, September 1999, pp. 1727–1736.

[8] J. J. Spilker, Jr. and A. J. Van Dierendonck, "Proposed New L5 Civil GPS Codes," *Navigation, Journal of the Institute of Navigation*, **48**(3), 135–144 (2001).

[9] J. J. Spilker, Jr., E. H. Martin, and B. W. Parkinson, "A Family of Split Spectrum GPS Civil Signals," *Proceedings of the 11th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1998)*, Nashville, TN, September 1998, pp. 1905–1914.

[10] B. C. Barker, J. W. Betz, J. E. Clark, J. T. Correia, J. T. Gillis, S. Lazar, K. A. Rehborn, and J. R. Straton, "Overview of the GPS M Code Signal," *Proceedings of the 2000 National Technical Meeting of The Institute of Navigation*, Anaheim, CA, January 2000, pp. 542–549.

[11] J.-L. Issler, L. Ries, J.-M. Bourgeade, L. Lestarquit, and C. Macabiau, "Probabilistic Approach of Frequency Diversity as Interference Mitigation Means," *Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2004)*, Long Beach, CA, September 2004, pp. 2136–2145.

[12] J. Betz, M. A. Blanco, C. R. Cahn, P. A. Dafesh, C. J. Hegarty, K. W. Hudnut, V. Kasemsri, R. Keegan, K. Kovach, L. S. Lenahan, H. H. Ma, J. Rushanan, J. J. Rushanan, D. Sklar, T. A. Stansell, C. C. Wang, and S. K. Yi, "Description of the L1C Signal," *Proceedings of the 19th International Technical Meeting of the Satellite*

Division of The Institute of Navigation (ION GNSS 2006), Fort Worth, TX, September 2006, pp. 2080–2091.

[13] GPS Directorate, Systems Engineering & Integration Interface Specification, Navstar GPS Space Segment/User Segment L1C Interface, IS-GPS-800, 21-Sept-2011, available at: http://www.navcen.uscg.gov/pdf/gps/IS-GPS-800B.pdf, visited July 14, 2012.

[14] United States Naval Observatory (USNO), GPS Operational Satellites, (Block II/IIA/IIR/IIR-M/II-F), 2012, ftp://tycho.usno.navy.mil/pub/gps/gpsb2.txt, visited July 14, 2012.

[15] FAA, GNSS Evolutionary Architecture Study, Phase I—Panel Report, February, 2008, available at: http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/library/documents/media/GEAS_PhaseI_report_FINAL_15Feb08.pdf, visited July 14, 2012.

[16] Global Navigation Satellite System (GLONASS), Interface Control Document L1, L2, Russian Institute of Space Device Engineering, Moscow, Version 5.1, 2008, available at: http://www.glonass-ianc.rsa.ru/en/, visited July 14, 2012.

[17] Septentrio, Septentrio's AsteRx3 Receiver Tracks First GLONASS CDMA Signal on L3, April 12, 2011, available at: http://www.insidegnss.com/node/2563, visited July 14, 2012.

[18] S. Revnivykh, "GLONASS Status and Modernization," *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2011)*, Portland, OR, September 2011, pp. 839–854.

[19] ESA, Galileo, 2012, available at: http://www.esa.int/esaNA/galileo.html, visited July 15, 2012.

[20] European Union, European Commission, Satellite Navigation, Galileo Open Service Signal-In-Space Interface Control, OS SIS ICD, Issue 1.1, September 2010, available at: http://ec.europa.eu/enterprise/policies/satnav/galileo/open-service/index_en.htm, visited July 15, 2012.

[21] China Satellite Navigation Office, BeiDou Navigation Satellite System, Signal In Space, Interface Control Document (Test Version), December 2011, available at: http://www.beidou.gov.cn/attach/2011/12/27/201112273f3be6124f7d4c7bac428a36cc1d1363.pdf, visited July 15, 2012.

[22] Japan Aerospace Exploration Agency, Quasi-Zenith Satellites System, Quasi-Zenith Satellite System Navigation Service, Interface Specification for QZSS (IS-QZSS), V1.4, February 28, 2012, available at: http://qz-vision.jaxa.jp/USE/is-qzss/DOCS/IS-QZSS_14_E.pdf, visited July 15, 2012.

[23] RTCA, Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment, DO-229C, RTCA, Washington, DC, November 2001.

5

GNSS ANTENNA DESIGN AND ANALYSIS

5.1 APPLICATIONS

While there are many global navigation satellite system (GNSS) receiver and antenna systems used today, they vary significantly in size, cost, capabilities, and complexity. These variations are largely driven by the end-user application. Clearly, the characteristics of a consumer-grade GNSS antenna are very different from those used in aviation, surveying, or space applications. A performance characteristic that is very critical for one application may not be important or even desirable in another application. The performance requirements for different applications are often mapped directly into the requirements for the antenna to be used in that particular application. The following sections will discuss several key antenna performance characteristics with discussion pertaining to their application.

5.2 GNSS ANTENNA PERFORMANCE CHARACTERISTICS

The overriding requirement of any GNSS antenna is to convert the various GNSS signals intended for use from an electromagnetic wave to an electrical signal suitable for processing by the GNSS receiver. To do this effectively, there are several important performance characteristics that must be considered in

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

the design of the GNSS antenna. Although there may be very specific performance characteristics for specific applications, the most important characteristics will be discussed here.

5.2.1 Size and Cost

In the world of antenna design, size (i.e., aperture) is arguably one of the most important factors. The eventual size of the antenna, most often constrained by the intended application, will often limit the eventual performance of the antenna. For example, the size and cost constraints of consumer cell phone applications are a major factor in the performance of the GNSS antenna within the cell phone. This size constraint is based upon the real estate allocated to the GNSS antenna considering all of the other functions to be performed by the cell phone and still allows it to fit in your pocket. For other applications, the size and cost may be less constrained so that other performance requirements can be achieved. For example, in a fixed ground-based GNSS reference station application, the antenna can typically be larger to produce high-quality code and carrier phase measurements but may cost more.

5.2.2 Frequency and Bandwidth Coverage

The GNSS antenna must be sensitive to the GNSS signals' center frequency and must have sufficient bandwidth to efficiently receive these signals, but there are several impacts on other performance characteristics that are important. Figure 5.1 graphically illustrates the various carrier frequencies and bandwidths for various GNSSs in the L-band (1.1–1.7 GHz).

As seen in Fig. 5.1, the various GNSS frequencies for a particular GNSS typically span a considerable amount (e.g., several hundreds of megahertz) so that ionosphere delay estimates can effectively be done for dual/multifrequency GNSS users. To have a single antenna span, the entire GNSS L-band from about 1150 to 1620 MHz would require an antenna to have about 33% bandwidth; this would likely require a broadband type of antenna design. Another consideration evident from Fig. 5.1 is at each center frequency corresponding to a particular GNSS signal, the bandwidth requirement is not that large when considering the carrier frequency (i.e., on the order of a couple percent bandwidth in most cases); these requirements could be met with a multifrequency band antenna design, which is typically different from a nearly continuous broadband antenna design. At first glance, the frequency and bandwidth requirements do not look very challenging, but when other performance requirements are considered for various applications, the design requirements become clearer.

The dual/multifrequency capabilities of various GNSSs tend to find application for more high-performance users who have requirements for comprehensive ionosphere error mitigation or, to a more limited extent, frequency

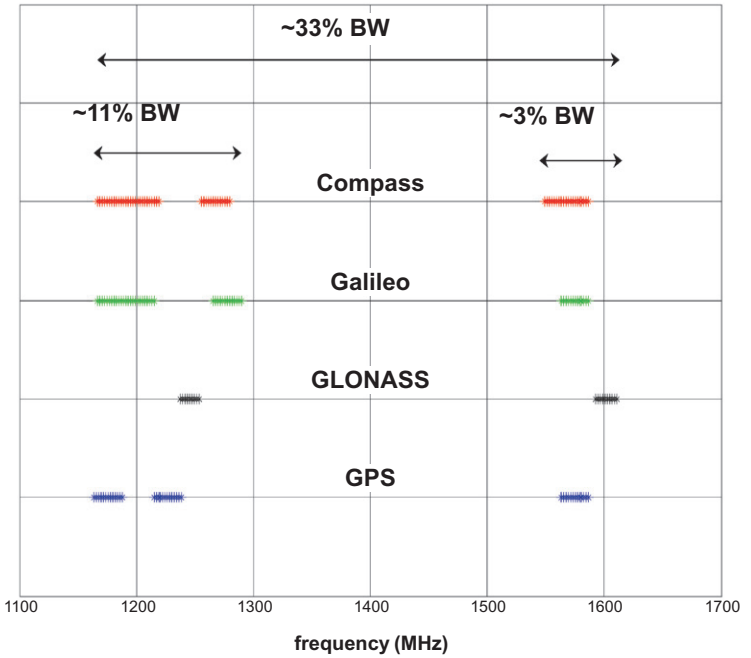


Fig. 5.1 Illustration of GNSS frequency and bandwidth (BW) for various GNSSs.

diversity application. Most other applications can be satisfied with the single-frequency capabilities inherent in a GNSS, such as low-cost consumer and general-purpose GNSS applications.

For example, to receive a GNSS signal at a single frequency over a relatively narrow bandwidth (e.g., L1, C/A-code), an antenna can typically be built much smaller and at a lower cost with less stringent performance requirements than a multifrequency wideband GNSS antenna (e.g., L1L2L5 and C/A, P(Y), M, L5). However, even for this single-frequency narrowband GNSS antenna, other performance requirements may dominate such as size and cost. Such applications in the cellular phone market demand small inexpensive antennas. As the size decreases, the efficiency will be decreased and special considerations for tuning become increasingly important. Furthermore, as the size of this type of antenna decreases, the radiation characteristics typically change.

For multifrequency GNSS antennas there are two general design approaches pursued. One approach is to design a multifrequency GNSS antenna with multiple resonances designed into the antenna structure. Such an antenna is typically a dual-frequency, dual-layer patch antenna. These types of antennas provide very good performance over a limited bandwidth but do not cover the bands between the frequencies of interest. The second general approach pursued in multifrequency GNSS antennas is a broadband design. Such a design would cover the entire band from the lowest frequency of interest to

the highest frequency of interest, where the performance may be optimized and specified for the various GNSS bands of interest. While the antenna performance may vary over the entire band, the performance at the frequencies of interest can easily be verified. Examples of these types of antennas are helix and spiral types of antennas. Helix antennas have historically been used on the Global Positioning System (GPS) and Global Orbiting Navigation Satellite System (GLONASS) space vehicles (SVs) and, in a more limited extent, in the user segment. With the advent of new GNSS signals and systems, the choice of broadband spiral-based antennas is becoming more popular in recent years for advanced user equipment.

5.2.3 Radiation Pattern Characteristics

The GNSS antenna must have sufficient gain to effectively convert the GNSS electromagnetic wave into a signal voltage so that it can be processed by the GNSS receiver. In accomplishing this goal, the GNSS antenna's radiation characteristic should exhibit certain characteristics. These will be addressed with respect to the desired and undesired signals of interest.

For a terrestrial GNSS user, the GNSS antenna should provide "nearly uniform gain" with respect to the GNSS SV elevation angle in the upper-hemisphere and omnidirectional coverage with respect to the GNSS SV azimuth angle. Now the description of nearly uniform gain means that while a constant gain of the GNSS SV signal from zenith down to the receivers mask angle is desirable, it is difficult to achieve and not explicitly required. Some gain variation can be tolerated in the antenna coverage volume, which will produce a variation in carrier-to-noise ratio (C/N_o) for the respective SV being tracked by the receiver; however, there is a limit to the amount of gain variation that can be tolerated in the GNSS receiver. This variation is typically limited by the code cross correlation within the GNSS receiver tracking loops. For GPS C/A-code processing, if one GPS pseudorandom noise (PRN) C/A-code is greater than another GPS C/A-code by about 20 dB, cross correlation can be significant. A couple of dB power level variations will occur due to the GNSS signal versus elevation angle variations, which will be part of this power budget. In most cases, a gain variation of no more than about 15 dB is desirable.

A mask angle used in GNSS antennas refers to an elevation angle (e.g., 5° or 10°) above the horizon, where a GNSS signal may be tracked by the receiver but is discarded. The reason for discarding signal measurements at these low elevation angles is that these signals often have more measurement error on them due to atmospheric effects, signal multipath, and lower C/N_o. For some geodetic applications, the mask angle is increased to 20° to help ensure high-quality measurements but at the expense of availability of the GNSS signals. Other applications may decrease the mask angle to help increase the GNSS signal availability. Still other systems, such as indoor applications, may not implement any mask angle due to the dynamic nature of the user antenna and the signal power challenged operational environment.

A significant number of user applications want to efficiently receive the desired GNSS SV signals of interest and at the same time minimize any undesired signals due to multipath or interference. Antenna design, ground planes, and pattern shaping can help receive the desired signals of interest and minimize the undesired signals of interest.

Most GNSS user antenna technologies involve a ground plane or planar structure so the GNSS antenna gain will naturally decrease as the elevation angles decrease. This is most often the dominant effect for decreasing C/N_o for low-elevation GNSS SV signal reception for terrestrial users. The use of ground planes for most applications additionally helps in the mitigation of multipath.

To optimize the received power from the GNSS antenna, the polarization of the antenna should match the polarization of the incident GNSS electromagnetic wave. Most radiation characteristics are depicted graphically in 2D or 3D plots where the pattern is represented as a far-field power pattern; 2D plots provide a better quantitative illustration of the radiation characteristics of the GNSS antenna and are often called an elevation (i.e., vertical) cut or an azimuth (i.e., horizontal) cut, but this depends upon the users' frame of reference.

For certain applications, the radiation characteristics of the GNSS antenna are less important than the requirement to make the antenna a small size or very inexpensive. An example of this type of application is in the mobile, low-cost consumer market. For these applications, the antenna must be a small size so that it can economically and practically fit within the size, cost, and real-estate constraints of the host device (e.g., a cell phone). In these types of applications, which include indoor applications, while multipath signals can cause large errors, they can also be used for degraded positioning if no direct signals are present. When the antenna size (i.e., aperture) is significantly decreased, the radiation pattern may have a more omnidirectional characteristic in all directions (i.e., isotropic) but will likely be affected by the other components around it.

5.2.4 Antenna Polarization and Axial Ratio

Although there is a wide variety of GNSS antennas, most are designed for right-hand circular polarization (RHCP) to match the polarization of the incoming GNSS signal. (The polarization of the GNSS electromagnetic signal is the direction the time-harmonic electric field intensity vector travels as the wave travels away from the observation point.) [1] For certain special cases, a GNSS antenna with polarization diversity or a linearly polarized (LP) antenna can be used. Polarization diversity can be used in some limited cases to help mitigate some known interference source, with known polarization. If an LP antenna is used to receive an RHCP GNSS signal and it is placed perpendicular to the incoming RHCP signal, 3-dB signal loss will occur due to the polarization mismatch; however, the gain of the antenna may be used to make up

for this polarization mismatch loss. As will be shown later, two orthogonally LP signal components can be combined to optimally receive a circularly polarized electromagnetic wave.

In general, the polarization of an electromagnetic wave is elliptical, with circular and linear polarization being special cases. The axial ratio (AR) of an antenna refers to the sensitivity of the antenna to the instantaneous electric field vector in two orthogonal polarization directions of the antenna, in particular, the magnitude of the maximum value (i.e., magnitude in the semimajor axis of the traced elliptical polarization) to the magnitude of the wave in the orthogonal direction (i.e., in the semiminor axis direction), as expressed in Eq. 5.1. Thus, for a GNSS antenna to be maximally sensitive to the incoming RHCP GNSS signal, it should have an AR of 1 (i.e., 0 dB), so that the electric field intensity vectors in the direction of maximum sensitivity are the same in the orthogonal direction. Depending upon the design type of the GNSS antenna, it is typically a good metric at boresight (i.e., in the zenith direction) for a GNSS antenna:

$$AR = \frac{E_{\text{major}}}{E_{\text{minor}}} = \frac{E_{\text{RHCP}} + E_{\text{LHCP}}}{E_{\text{RHCP}} - E_{\text{LHCP}}}, \tag{5.1}$$

where

E_{major} = magnitude of **E** in the direction of the semimajor or axis
E_{minor} = magnitude of **E** in the direction of the semiminor or axis
$E_{\text{RHCP}} = \dfrac{(E_\theta + jE_\phi)}{\sqrt{2}}$
$E_{\text{LHCP}} = \dfrac{(E_\theta - jE_\phi)}{\sqrt{2}}$
E_θ = complex E field in the spherical coordinate direcion θ
E_ϕ = complex E field in the spherical coordinate direcion ϕ
$\boldsymbol{E}$ = $E_\theta \boldsymbol{a}_\theta + E_\phi \boldsymbol{a}_\phi$
$\boldsymbol{a}_\theta$ = unit vector in the θ direcion
$\boldsymbol{a}_\phi$ = unit vector in the ϕ direcion

To illustrate how RHCP (and left-handed circular polarization [LHCP]) can be produced, consider two orthogonally placed dipole antennas as shown in Fig. 5.2 from a "top-view" perspective receiving an incident RHCP signal at zenith (i.e., straight into the page). The center of the two dipoles is at the origin of the local antenna coordinate system in the x-y plane, where dipole #1 is aligned with the x-axis, dipole #2 is aligned with the y-axis, and the z-axis is pointed outward (i.e., out of the page).

Now, when the GNSS signal leaves the satellite, it can be described as propagating in a $+z_{\text{SV}}$ direction (in an SV coordinate system) and represented

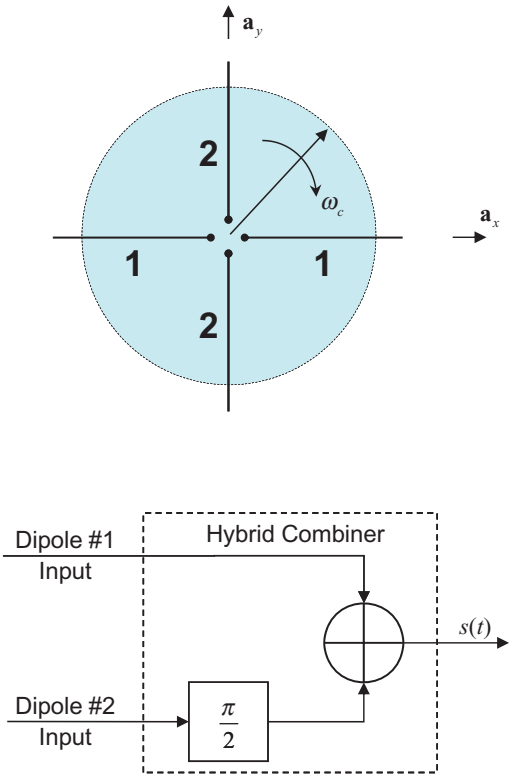


Fig. 5.2 Antenna configuration for reception of a GNSS RHCP signal.

as a normalized GNSS RHCP signal as $\mathbf{E}_{\text{GNSS}}(z_{\text{sv}}, t) = \cos(\omega_c t)\mathbf{a}_x + \sin(\omega_c t)\mathbf{a}_y$. This GNSS signal is incident on the GNSS antenna, where the antenna coordinate system has a + z direction pointed toward zenith (i.e., in the opposite direction of the SV coordinate system $+z_{\text{SV}}$). Thus, we can represent the incident normalized GNSS signal onto our local antenna coordinate systems, as shown in Fig. 5.2, as $\mathbf{E}_{\text{GNSS}}(z, t) = \cos(\omega_c t)\mathbf{a}_x - \sin(\omega_c t)\mathbf{a}_y$.

The normalized signal response on dipole #1 due to the incident GNSS signal $E_{\text{GNSS}}(z, t)$ can be represented as $s_1(t)$, and normalized signal on dipole #2 due to the incident GNSS signal can be represented as $s_2(t)$, where these two signals can be simply represented as shown in Eq. 5.2:

$$\begin{aligned} s_1(t) &= \cos\omega_c t \\ s_2(t) &= \sin\omega_c t. \end{aligned} \tag{5.2}$$

For the incident RHCP GNSS signal, we delay the leading E field term and then combine the signals to coherently add the signal response from each dipole. Adding the 90° delay to dipole #2, as shown in Fig. 5.2, combining, then

the combined signal can be represented as Eq. 5.3. When the response from the two orthogonal components is the same, then the *AR* will be equal to 1 (i.e., 0 dB):

$$s(t) = \cos\omega_c t + \sin\left(\omega_c t + \frac{\pi}{2}\right) = 2\cos\omega_c t. \tag{5.3}$$

If the incident signal was LHCP incident onto the antenna, as described by $\mathbf{E}(z, t) = \cos(\omega_c t)\mathbf{a}_x + \sin(\omega_c t)\mathbf{a}_y$ in the antenna coordinate system, then the signal received would be naturally 90° lag in the opposite direction and, with the 90° phase shift in the hybrid combiner, would produce an output response where the net signal from dipole #1 and dipole #2 would cancel, producing a zero response at the output.

Depending upon the design type of the GNSS antenna, the *AR* is typically a good metric at boresight (i.e., in the zenith direction) for a GNSS antenna, but the *AR* will typically get bigger as the elevation angle decreases. While this does depend on the antenna design, installations, and application, consider the following. Patch antennas are typically mounted on ground planes to improve their radiation characteristics, to minimize impedance variations, and to mitigate multipath. A well-known boundary condition in electromagnetics is that the electric field tangent to a perfect electric conductor will go to zero. As the elevation angle decreases, the electric field component of the incoming GNSS signal will obey the boundary condition and, subsequently, the "horizontal component" (i.e., the component parallel to the ground plane) will go to zero. This will leave the vertical component (i.e., the component perpendicular to the ground plane) as the dominant component. This is one of the reasons why GNSS antennas mounted over ground planes have a gain reduction as the elevation angle decreases.

5.2.5 Directivity, Efficiency, and Gain of a GNSS Antenna

The directivity of an antenna is the ratio of the radiation intensity in a given direction, normalized by the radiation intensity averaged over all space [2]. For GNSS applications, often the shape (i.e., directivity) of the antenna radiation characteristics is very important to help maintain a more constant received signal power level for the various GNSS signals being tracked at various aspect angles from the antenna. Too much gain variation (e.g., >~18 dB or so) across the coverage region of the GNSS satellite signals, can cause significant cross-correlation issues in the code tracking loops between different PRNs. The amount of cross-correlation interference will depend upon the code type, frequency difference, and power level differences between the signals.

The gain of a GNSS antenna is related to the directivity, by the antenna efficiency in accordance with Eq. 5.4 [3]:

$$G(\theta, \phi) = e_{cd} D(\theta, \phi) \quad \text{(unitless)}, \tag{5.4}$$

where

$$D(\theta,\phi)=\frac{U(\theta,\phi)}{U_{\text{AVG}}},$$

$$e_{cd}=\text{efficiency}=\frac{P_{\text{rad}}}{P_{\text{in}}}$$

P_{rad} = power radiated by the antenna (W)
P_{in} = power input to the antenna (W).

The gain for an antenna in Eq. 5.4 is often stated in units of dB (technically dBi [dB relative to an isotropic (i) radiator]), for a particular polarization. As for the efficiency of the antenna, the input power specified in Eq. 5.4 represents the power into the actual antenna terminal, and with the theory of reciprocity, this power can be viewed as the output power, at the antenna terminals for a GNSS reception antenna; likewise, power radiated can be viewed as power incident. For passive GNSS antennas, the gain is nominally 0 dBi over the upper hemisphere, but will often be slightly greater at zenith (e.g., +3 dBi), and lower (e.g., –3 dBi) at low elevation angles (e.g., 80° away from zenith). For electrically large or phased-array antennas the gain can increase substantially above the nominal 0 dBi, and care must be taken in providing sufficient gain to the desired GNSS signals to be received while suppressing undesired signals for reception to help minimize multipath and/or interference sources.

For GNSS antennas that are integrated with other radio frequency (RF) components, including active amplifiers, often the overall gain of the antenna is specified to include these devices. This is especially true for hermetically sealed active GNSS antennas that are designed, fabricated, and sold as a single hermetically sealed package.

GNSS antennas with active components are typically supplied with a DC voltage from the GNSS receiver up the RF transmission line. Coupling between the DC and RF GNSS signals is handled on either end with a "bias-T" that separates the DC and RF signals. Typically, the voltages vary from 3 V to upward of 18 V and are often accompanied with a voltage regulation circuit to maintain constant gain in the active amplifiers and provide for some versatility when connecting various antenna and receiver combinations.

5.2.6 Antenna Impedance, Standing Wave Ratio, and Return Loss

Three additional performance parameters for a GNSS that can *help* assess the performance of a GNSS antenna are the antenna's "input" impedance, standing wave ratio (SWR), and return loss (RL). Emphasis is placed on the word *help* because these parameters should not be used in a vacuum and are not exclusive. Just because a device has good impedance, a low SWR, and high RL does not exclusively mean it is a good antenna for the application; we must

still look at the radiation pattern and other performance characteristics (e.g., a 50-Ω load connected to a 50-Ω transmission line is a perfect match, has a nearly ideal SWR, and high RL, but is not a good antenna).

The theory of reciprocity in antenna theory states that, from a passive antenna perspective, the performance characteristics of the antenna will be same, whether we think of the antenna from a transmission or reception perspective. Thus, we can consider the antenna's "input" impedance (from a transmission perspective), the same as the "output" impedance (from a reception perspective), at the antenna terminals (Z_A). A desired characteristic of the GNSS antenna is to match the antenna impedance to the impedance of the transmission line that is connected to it. Most often, a transmission line with a characteristic impedance (Z_o) of 50 Ω is used. With knowledge of the transmission line characteristic impedance and the antenna impedance, the reflection coefficient (Γ_A) at the antenna terminal can be calculated as shown in Eq. 5.5. The reflection coefficient will vary from –1 to +1, where 0 is the ideal case under impedance match conditions. The reflection coefficient can then be used to calculate the SWR and match efficiency, as shown in Eq. 5.5. In the ideal case, the $\Gamma_A = 0$, SWR = 1.0, and $e_r = 1$. Some engineers prefer to use the RL when characterizing the impedance match of the antenna in dB format, which is again shown in Eq. 5.5. For example, if the RL is 20 dB, that means that the signal that gets reflected at the antenna terminal location is 20 dB down from the incident signal:

$$\Gamma_A = \frac{Z_A - Z_o}{Z_A + Z_o} = \frac{SWR - 1}{SWR + 1} \quad \text{(unitless)}$$
$$\text{SWR} = \frac{1 + |\Gamma_L|}{1 - |\Gamma_L|}, \text{ where } 1 \le \text{SWR} \le \infty \quad \text{(unitless)}, \tag{5.5}$$

where

Γ_A = reflection coefficient at the antenna terminal (unitless)
Z_0 = character istic impedance of the transmission line (Ω)
Z_A = antenna impedance (i.e., at the treminal) (Ω)
e_r = mismatch (i.e., reflection) efficiency = $(1 - |\Gamma_A|^2)$ (unitless)
RL = return loss = $-20 \log_{10} |\Gamma_A|$ (dB).

5.2.7 Antenna Bandwidth

The bandwidth of a GNSS antenna refers to the range of frequencies where the performance of the antenna is satisfactory with respect to a particular performance metric [2]. For a single-frequency GNSS antenna, the range of frequencies is usually centered around the carrier frequency and must be wide enough to provide the downstream receiver system with enough signal fidelity

so that the GNSS signal can effectively be processed. For example, in a low-cost C/A-code GPS application, the first null-to-null bandwidth of the signal (i.e., 2×1.023 MHz) is often used for a minimum bandwidth requirement. Other applications could be narrower with some signal power reductions. For other high-performance applications that use an advanced receiver design (e.g., narrow correlator), a wider bandwidth is needed to enable more effective tracking of the signal in the receiver [4]. Such applications may require up to 16 MHz in bandwidth to effectively track the C/A-code and gain the advantages provided by the advanced correlator design in the receiver.

For dual or multifrequency antennas that cover distinct bands (e.g., L1 and L2), it is appropriate to talk about the bandwidth of the antenna for each band. Certain antenna technologies lend themselves well to this type of bandwidth characterization, such as patch antennas that are typically narrowband at a given resonance frequency, but can be layered or stacked to support multiple frequency bands.

As the number of GNSSs increases, including an increase in the number of frequency bands, a more "broadband GNSS signal" design approach could be taken for the GNSS antenna. Certain antenna technologies can be selected to design a GNSS antenna that would cover the entire band from 1100 to 1700 MHz to provide sufficient antenna bandwidth to efficiently receive the GNSS signals regardless of their specific frequency in that 1100- to 1700-MHz band. These types of antennas are becoming increasingly popular as the number of GNSSs increases. A spiral antenna design is a good example of this type of antenna that could cover the entire GNSS band from 1100 to 1700 MHz.

As stated earlier, the bandwidth of a GNSS antenna refers to the range of frequencies where the performance of the antenna is satisfactory with respect to a particular performance metric. For various applications, the particular performance metric for satisfactory antenna performance may be specified in different ways. This satisfactory performance can refer to the gain, antenna radiation pattern characteristic, polarization, multipath performance, impedance, SWR, or some other metric. Now, some of these performance metrics are more complicated to measure and/or quantify versus frequency. A simple and easy to measure metric is the SWR. Often a maximum value for the SWR will be specified to help characterize the bandwidth. So as long as the SWR is less than that specified maximum value over a range of frequencies that cover the desired signal frequency and satisfactory performance is achieved, the upper and lower limits of the maximum SWR values can be used to calculate a bandwidth. Figure 5.3 illustrates the SWR of a typical GNSS antenna plotted from 1 to 2 GHz with the 2.0:1.0 SWR bandwidth metric indicated.

For broadband antennas, often the SWR is not the best metric to characterize the bandwidth of the antenna. For example, a helix antenna will typically have a reasonably good SWR over a large bandwidth; however, the antenna radiation pattern characteristics will typically deteriorate before the SWR metric goes bad.

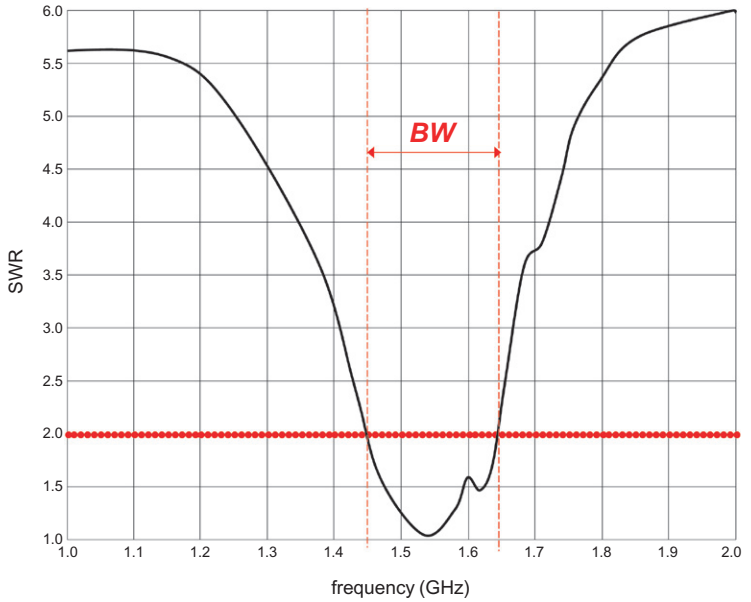


Fig. 5.3 Illustration of a GNSS antenna bandwidth using a 2.0:1.0 SWR metric.

5.2.8 Antenna Noise Figure

The noise figure (*NF*) of a device refers to the amount of noise that is added to the output with respect to the input. The *NF* can be expressed as the ratio of the input signal-to-noise ratio $(S/N)_{in}$ to output signal-to-noise ratio $(S/N)_{out}$, typically expressed in units of dB, as shown in Eq. 5.6 [5]:

$$NF = 10\log_{10}(F) \quad \text{(dB)}, \tag{5.6}$$

where

$$F = \frac{(S/N)_{in}}{(S/N)_{out}} = \text{noise factor (power ratio) (unitless)}$$

$$T_{device} = T_0\,(F - 1)$$

T_0 = reference temperature
= 290 K (US)
= 293 K (Japan).

As shown in Eq. 5.6 the *NF* is often stated as noise factor, a unitless power ratio, which can be related to an actual device temperature (T_{device}).

For a GNSS antenna element, two factors that will affect the antenna *NF* are the antenna brightness temperature and the physical temperature of the

antenna. The antenna brightness temperature is related to what the antenna is "looking at." For example, the noise at the output of the antenna terminals will be different if the GNSS antenna is operated in an outdoor environment, as opposed to inside an anechoic test chamber. The physical temperature of the antenna relates to how hot or cold, physically, the antenna is. For certain space and/or missile applications, where the antenna can see large variations in its physical temperature, the *NF* variations may be a consideration factor affecting the overall performance of the antenna/receiver systems.

Depending upon the application, GNSS antenna elements may be designed and operated as "passive" or "active" devices. Passive GNSS antennas are those that have GNSS antenna elements and no other active devices (i.e., amplifiers) integrated within the antenna enclosure. (On occasion, a passive bandpass filter [BPF] may be integrated within an antenna enclosure and is referred to as a passive GNSS antenna.) An active GNSS antenna refers to a GNSS antenna configuration where active amplifiers are included within the GNSS antenna enclosure. Often direct access to the antenna connections (i.e., antenna terminal) is not possible because of packaging and the desire to minimize cost, reduce size, or to hermetically seal the unit for environmental considerations. For active antennas in an integrated package, independent measurements of the gain and noise of the antenna itself are difficult to measure because direct access to the antenna terminals is not possible. An alternative approach is to measure the gain normalized by the equivalent noise temperature (G/T) of the device. This type of technique has been used in the satellite communications community for a number of years and similar techniques have been established by RTCA, Inc. for the performance characterization of active integrated GPS antennas [6].

For most general-purpose applications with active GNSS antennas, the *NF* is not a major factor in systems performance but should be considered for certain applications. For example, applications of GNSS antennas on fast-moving missiles/projectiles, high-sensitivity, indoor applications, and rooftop/lab installations require special attention. While a detailed *NF* chain analysis is beyond the scope of the material presented here, any passive loss between the passive antenna and the first active device in the receiver chain will directly add to the *NF*. Thus, caution should be taken when a passive antenna is used, for example, on a rooftop installation, where the first active amplifier is placed at a substantial distance from the passive antenna element. Placing a high gain, low-noise amplifier (LNA) device close to the antenna terminal output is good practice to help minimize the overall receiver system *NF*.

5.3 COMPUTATIONAL ELECTROMAGNETIC MODELS (CEMs) FOR GNSS ANTENNA DESIGN

Historically, antenna design has been based on the analytical foundations of electromagnetic theory that has led to physical designs, fabrication, test, and

often iteration to achieve satisfactory performance. In the past, the iteration step typically involved physically constructing the antenna, testing, then adjusting the physical design, and retesting. This iteration loop continued until satisfactory performance was achieved. With the advancements in recent years in computer technology and CEMs, the antenna design interaction loop is largely done in simulation. While there may still need to be physical design iterations, the number of physical designs can be significantly reduced with the implementation of CEMs in the antenna design process.

CEMs have become increasingly popular due to the increased efficiency, cost savings, and shorter time to market for antenna designs. For GNSS antenna design, CEMs that are based on "numerical methods" are well suited because the physical size of the GNSS antenna is not significantly larger than the wavelength of the GNSS signal. CEMs solve for the electric (E) and magnetic (H) fields across a "grid" over a region of space based on Maxwell's equations (integral form or differential form). Furthermore, the solutions of E and H can be computed in the time domain or frequency domain. Various companies and agencies continue to develop CEMs and have even combined various CEM techniques to provide software packages and products that are better suited for a wide variety of problems. In the paragraphs that follow, a few of the CEMs and methods will be discussed that are useful for GNSS antenna design.

The procedure of using CEMs typically involves many steps in defining the antenna to obtain performance predictions. For antenna design, the first step involves building a geometric model of the antenna and any surrounding objects to be simulated. The initial antenna design is based on the theoretical analytical design of the antenna. If nearby parts or objects are desired to be included in the simulation, they, too, would be geometrically built into the simulation model. Next, the electrical material properties of the components in the model need to be specified, including the conductivity (σ), the permittivity (ε), and permeability (μ). Next, the excitation on the antenna would be produced within the model. From the theory of reciprocity, this can be thought of as a simulation source, and the performance will represent the antenna as a transmission or reception antenna (assuming no active or unidirectional devices are built into the model). Most CEMs include the ability to provide a "port excitation" or a "wave port" excitation, that is, from a voltage or current on a conductor, or an electromagnetic wave propagating into the simulation region of space. CEMs solve for the E and H fields across the domain of the model in "small pieces or cells" using a geometric grid-type structure. These computations are done across the grid in the model. Some CEMs will automatically compute the grid and adapt the grid based on the frequency, span of frequencies, shape, and material properties of the components in the model. Once the E and H fields have been calculated across the grid, performance predictions of the antenna in its simulated surroundings can be computed. Performance parameters such as radiation characteristics, impedances, and SWR can readily be produced.

One of the most popular CEM methods implements a "method of moments (MoM)" technique that solves for current distributions and the *E* and *H* fields using the integral format of Maxwell's equations. MoM techniques are very well suited for solving antenna problems that involve wire and wire-based array antenna designs. MoM techniques have also been called Numerical Electromagnetic Code (NEC) as originally developed to support the U.S. Navy. Today, various companies and agencies continue to develop the core computation engine (i.e., the NEC) and have graphical user interfaces (GUIs) integrated with the NEC to provide a user-friendly interface. Popular CEMs that implement MoM techniques are FEKO [7], IE3D [8], and WIPL-D [9].

Finite element methods (FEMs) typically use a triangular cell structure and solve for the *E* and *H* fields using a differential equation-based implementation of Maxwell equations. FEMs have found applications in scattering and patch antenna design applications and are often integrated with other CEM techniques to provide enhanced capability. One popular CEM that implements FEM techniques is the High Frequency Structure Simulation (HFSS) [10].

One of the earliest introduced techniques of CEMs is the finite difference time domain (FDTD) method. With the advances in computer computation capability, FDTD methods have become more efficient in recent years. FDTD methods solve for the *E* and *H* fields in a "leap frog" method across the grid in time steps. FDTD methods are well suited for broadband computations, but computation time will increase as the model becomes bigger with respect to the simulated wavelength. One popular CEM that implements FDTD techniques is the XFdtd® [11].

5.4 GNSS ANTENNA TECHNOLOGIES

5.4.1 Dipole-Based GNSS Antennas

One of the most fundamental electrically sensitive antennas is a half-wave dipole antenna that operates near resonance of the desired frequency of operation. Dipole antennas are sensitive to electric fields that are colinear with the orientation of the dipole. The length is one-half of the respective wavelength and is tuned to produce a good impedance match. An ideal $\lambda/2$ dipole will have an impedance of $Z_A = 72 + j42.5\,\Omega$. Typically, the length is shortened slightly to reduce the real part of the impedance to $50\,\Omega$ and a small amount of capacitance could be added to cancel out the slight inductance of the reactive part of the antenna impedance. As an antenna size decreases less than half-wavelength, the efficiency and gain of the antenna will decrease.

5.4.2 GNSS Patch Antennas

One of the most popular GNSS antenna types is the patch antenna. Patch antennas offer many advantages and a couple of disadvantages for various

Fig. 5.4 Patch antenna aviation form factors (with radome) [13] (courtesy of Sensor Systems®, Chatsworth, California).

applications. Patch antennas are a form of microstrip antennas that are typically printed on the surface of a microwave dielectric substrate material with a ground plane on the bottom side of the dielectric substrate material. Patch antennas date back to the 1950s and today have wide spread applications in GNSS [12]. They have a low profile, which is a significant advantage for dynamic vehicles to minimize wind resistance, snow/ice buildup, and minimize breakage. The low profile is also an advantage in consumer applications such as cell phones. The low profile lends itself well to high-volume production methods and integration with RF front-end circuits such as LNAs and BPFs that can be added to the back side (bottom of the antenna ground plane) of the antenna. Patch antennas can be produced in various form factors to suit a wide variety of applications from low-cost consumer products to high-performance aviation markets. Figure 5.4 illustrates a commercially available aviation patch antenna [13], conforming to the aviation ARINC 743A [14] form factor.

The microstrip patch antennas start with a high-quality material that is often double-clad copper on the top and bottom, whereby some of the copper on the top is etched/removed to form the radiating element of the patch. While patch antennas can be fabricated with techniques similar to printed circuit boards, the dielectric materials that make up the substrate as well as the fabrication process need to be tightly controlled to help ensure acceptable performance over a variety of temperature and operational environments. Higher-performance dielectric materials are made from specific dielectric compounds to help minimize variations in electrical properties during manufacturing and over various operational temperatures. In particular, the relative permittivity (ε_r) and the dielectric thermal expansion coefficient are controlled for the dielectric substrate materials. The thickness of the dielectric substrate

materials is a small fraction of the wavelength, (e.g., 0.5–5.0% [i.e., ~1 to 10 mm]) with ε_r typically in the 2–10 region. There are various manufacturers that make these types of high-frequency dielectric materials such as Rogers [15] and Taconic [16].

Microstrip patch antennas operate in a cavity resonance mode where one or two of the planner dimensions of the patch are of lengths equal to one-half of the wavelength in the dielectric substrate material (i.e., $\lambda_d/2$). With the patch element length equal to $\lambda_d/2$, the cavity will resonate at its fundamental or dominant frequency. It should be kept in mind that higher-order modes will also resonate, but these modes often produce undesirable performance characteristics for GNSS antennas (e.g., undesirable radiation pattern characteristics). There are a variety of shapes that can be used for patch antennas including square, nearly square, round, and triangular shaped. Additionally, there are a variant of feeding techniques for patch antennas including edge fed, probe fed, slot fed, and other variants. Configurations to provide a foundation for patch antenna design, and those commonly found in GNSS antennas, will be discussed next.

5.4.2.1 Edge-Fed, LP, Single-Frequency GNSS Patch Antenna To provide a foundation for patch antenna design, consider a single-frequency, edge-fed patch antenna as shown in Fig. 5.5, illustrated for two orthogonal orientations. The patch is edge and center fed and will produce LP for the individual orientations as illustrated by the magnitude of the far-field electric field vector $|\mathbf{E}^{FF}|$. For patch antennas, radiation comes from the edges of the patch as illustrated in Fig. 5.5 between the top conductive patch and the bottom ground plane. The dielectric substrate is between the two conductive materials. To obtain a resonance frequency f_r, the length (L) of the patch can be set to one-half of the wavelength in the dielectric material with relative permittivity (ε_r) in accordance with Eq. 5.7. To account for the fringing of the fields at the edges of the patch, the effective length of the patch can be increased slightly; an approximation is provided in Eq. 5.7:

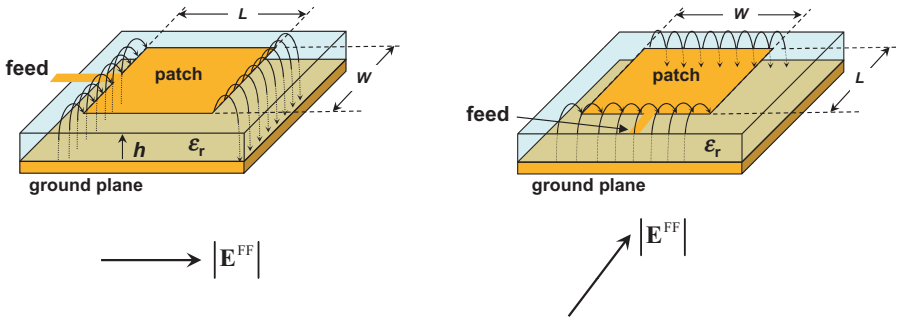


Fig. 5.5 Illustration of a single-frequency edge-fed patch antenna.

$$L = \frac{\lambda_d}{2} = \frac{\lambda_r}{2\sqrt{\varepsilon_r}} = \frac{c}{2f_r\sqrt{\varepsilon_r}}, (\text{m})$$
$$\text{To account for fringing:}$$
$$L \rightarrow L_{\text{eff}} \approx L + 2h \tag{5.7}$$

where

L = length of patch design for resonance, (m)
L_{eff} = effective length of patch design for resonance, (m)
λ_d = wavelength in dielectric, (m)
λ_r = wavelength for desired resonance, (m)
ε_r = relative permittivity of dielectric substrate, (unitless)
c = speed of light, (m/s).

The radiation from the edges of the patch antenna illustrated in Fig. 5.5 act like slot radiators. When the length of the patch is adjusted "tuned" for the desired resonance frequency, at a particular instant of time, the E field will come from the ground plane to patch at radiating slot #1 (along feed edge) and from the top patch to the ground plane at radiating slot #2 (along far edge). Internal to the substrate (between the top patch and the ground plane), the E field will decrease in amplitude, moving toward the center of the patch. In the middle of the patch, the E field will be zero.

For this simple edge-fed patch, Jackson and Alexopoulos computed the approximate formulas for the input resistance and bandwidth [17] as shown in Eq. 5.8. These expressions provide good insight that an increase in bandwidth can be achieved by increasing the height of the dielectric material or increasing the width (W) of the top radiating element. Here, the bandwidth is shown using an SWR less than 2.0 metric and expressed as a fractional bandwidth (i.e., the bandwidth with respect to the resonance frequency). The impedance will be maximum at the edge of the patch and will decrease in a sinusoidal fashion to a value of 0 Ω in the middle of the patch in accordance with Eq. 5.8, where y_o represents the offset distance from the edge, along the center line:

$$Z_{\text{in}} = 90\frac{\varepsilon_r^2}{\varepsilon_r - 1}\left(\frac{L}{W}\right)^2, (\Omega)$$
$$\text{BW}_{\text{SWR=2:1}} = 3.77\left(\frac{\varepsilon_r - 1}{\varepsilon_r^2}\right)\left(\frac{W}{L}\right)\left(\frac{h}{\lambda}\right), (\text{fractional}), \tag{5.8}$$

where

$$\frac{h}{\lambda} << 1$$

and

$$Z_{\text{in}}(y = y_{\text{o}}) = Z_{\text{in}}(y = 0)\cos^2\left(\frac{\pi y_{\text{o}}}{L}\right)(\text{ohms}),$$

where

y_{o} = offset distance from the edge along the center line, (m).

5.4.2.2 Probe-Fed, LP, Single-Frequency GNSS Patch Antenna Probe-fed GNSS antennas are extremely popular and have the advantage of minimizing the overall size of the antenna and integrating the RF front-end amplifiers and filters on the back side of the antenna ground plane. Figure 5.6 illustrates a passive patch antenna that is probe fed with a coaxial cable or connector output. The center conductor is connected to the top patch via a hole in the dielectric material (and ground plane), and the connector ground/return is connected directly to the bottom ground plane. Once again, the probe can be connected to the edge or moved inward to help match the antenna impedance to the cable/connector impedance [18, 19]. A factor that should be considered with probe-fed patch antennas is the inductance introduced by the probe going through the dielectric substrate at the desired resonance frequency [20]. The real and imaginary parts of the input impedance for a probe-fed patch antenna can be analytically approximated shown in Eq. 5.9:

$$R_{\text{in}}(y = y_{\text{o}}) \cong 60\frac{\lambda_{\text{r}}}{W}\cos^2\left(\frac{\pi y_{\text{o}}}{L}\right) \quad (\Omega)$$

and (5.9)

$$X_f \approx -\frac{\eta kh}{2\pi}\left[\ln\left(\frac{kd}{4}\right) + 0.577\right] \quad (\Omega),$$

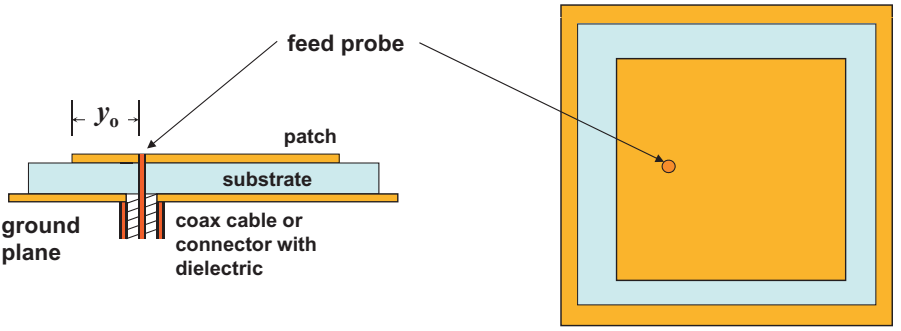


Fig. 5.6 Passive probe-fed patch antenna with a coaxial/connector output.

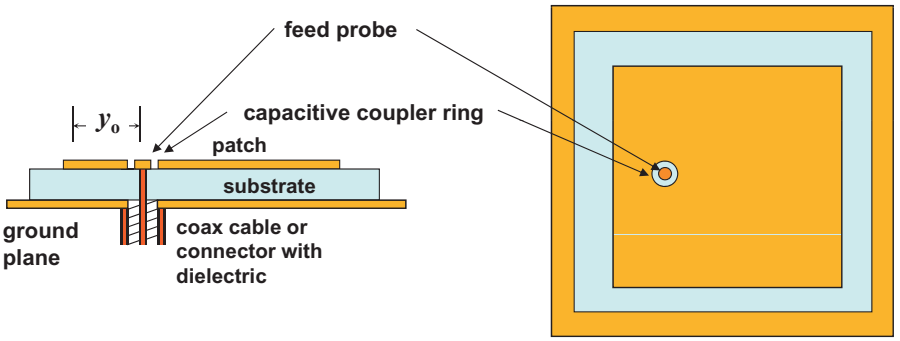


Fig. 5.7 Probe-fed patch antenna with capacitive coupler ring.

where

η = intrinsic impedance, (ohms)
k = phase constant, (rad/m)
d = diameter of probe, (m).

For some low-cost, low-bandwidth (e.g., C/A-code) applications, a thin substrate can be used and the added inductance can be tolerated; however, for high-quality and larger-bandwidth patch antennas, this added inductance can be compensated for. As shown in Eq. 5.9, as the height of the dielectric substrate increases, so does the inductance at the feed. There are several approaches that can be implemented to help compensate for the probe-feed inductance. One approach is to tune the patch at a slightly higher frequency (slightly above the resonance frequency), where the probe inductance will be close to zero. Another popular approach is to add a small amount of capacitance to the feed. Once again, there are several ways to add capacitance to the feed location; Fig. 5.7 illustrates a common capacitive coupler ring approach. This approach simply adds a thin circular ring where the electromagnetic signal is coupled between the probe and the top patch element [21, 22]. The width of this ring is typically small (e.g., <1 mm) and will help cancel out the probe inductance.

5.4.2.3 Dual Probe-Fed, RHCP, Single-Frequency GNSS Patch Antenna

Each of the patch antennas presented above, edge-fed or probe-fed, produces (or receives) LP. In the designs presented above, the feed, either edge fed or probe fed, was placed in the center of the feed side (i.e., at $W/2$). To generate or receive circular polarization, each of these geometrically orthogonal signal polarization components can be combined, in a quadrature fashion, to produce either RHCP or LHCP. The combination of the two orthogonal signal components can be combined at RF using an RF hardware device such as a 90°

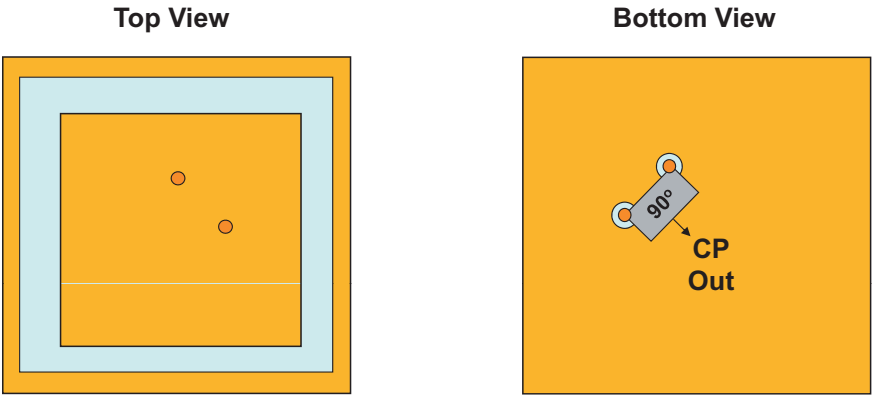


Fig. 5.8 Dual probe-fed patch antenna with separate quadrature power combiner (top and bottom view).

microstrip hybrid combiner printed directly on the antenna substrate or a separate RF circuit component (i.e., similar 90° hybrid power combiner); this approach will produce an RF output that will be RHCP or LHCP, depending upon the port combination. For RHCP, the leading port from a top-view signal reception perspective, will be delayed so that the two ports get combined in phase. Impedance matching of each port can be performed within the microstrip hybrid combiner for an edge-fed patch or by moving the probe feed inward from the edge. The size of each side (i.e., L) of the patch will be tuned to obtain resonance at the desired frequency; thus, the patch can be square (i.e., perfectly square). Figure 5.8 illustrates this dual probe-fed square patch for a single-frequency GNSS antenna. Here, each port can be used individually to receive each LP component or combined in quadrature for RHCP or LHCP. (Form RHCP in accordance with Eq. 5.2.)

5.4.2.4 Single Probe-Fed, RCHP, Single-Frequency GNSS Patch Antenna
In all of the above LP cases, the antenna feed was placed in the center of the feed side. As discussed earlier, as the feed is moved inward, from the edge toward the center, the impedance will decrease; this procedure can be used to help match the input impedance of the antenna to the transmission line. If, however, the feed location is moved laterally (i.e., away for the center of the feed side to the left or right), then the polarization will go from being purely linear to elliptical, and possibly circular toward the diagonal of a patch antenna that is "nearly square." For an edge-fed patch, the feed can be placed directly on the corner and the impedance matched. For a probe-fed patch, the feed is most often placed on the diagonal of the nearly square patch and moved inward, away from the corner to match the impedance.

The selection of the L and W of the nearly square patch antenna is important to obtain good CP performance. When the feed is placed on or near the

diagonal of the patch antenna, two polarization or transmission modes will be produced internal to the patch (i.e., in the dielectric medium, that can be represented as a resonance cavity, but now with two orthogonal signals within). The key is to provide both of the two orthogonal components, at the feed location, with equal amplitude, where one signal is 45° leading a reference phase, and the other signal is 45° lagging the reference phase. Since the two signal component phases are different at the feed location, their respective resonance frequency will be different for each of the two orthogonal signals, and the reference phase should be related to the desired resonance center frequency. (Often the two internal signal components are referred to as degenerative modal signals.) The corresponding lengths of each side of the patch will be slightly different, to produce two orthogonal degenerative modal signals, with slightly different resonance frequencies. The net effect is to produce two equal amplitude signals that are 90° out of phase, at the feed location, that will effectively add to produce an RHCP (or LHCP) signal at the feed location.

The total quality factor (Q_t) for the antenna at resonance can be used as a basis for calculating the difference between the lengths of each of the sides of a nearly square patch antenna [23]. The selection of the two lengths of the nearly square patch can be designed in accordance with Eq. 5.10, where one design degenerative mode, f_{10}, is set below the desired resonance frequency (f_r), and the other design degenerative mode, f_{01}, is set to be above f_r, by a nearly equal amount. Both Q_t and the AR will go into the approximation of the BW as shown in Eq. 5.10:

$$L = W\left(1 + \frac{1}{Q_t}\right)$$

and

$$\begin{aligned} f_{10} &= \frac{f_r}{\sqrt{1 + 1/Q_t}} \\ f_{01} &= f_r\sqrt{1 + 1/Q_t} \\ BW &\approx \frac{12AR}{Q_t} \quad \text{(percent)}, \end{aligned} \tag{5.10}$$

where

$$L = \frac{\lambda_{dL}}{2}$$

$$W = \frac{\lambda_{dW}}{2}$$

Q_t = quality factor at resonance for the antenna
AR = axial ratio.

Consider the following example. For a Q_t equal to 10, a total of 10% variation in the two degenerative signal modes (i.e., f_{10} and f_{01}) will be desired. Thus, f_{10} could be selected as 5% below f_r and f_{01} could be selected to be 5% above f_r. The lengths of each side of the nearly square patch would be different by about 10%. At boresight, if the AR is 1, then the percent bandwidth would be 1.2% or about 16 MHz at the GNSS L1 frequency.

Now, as described above, the corresponding lengths of each side of the patch would be slightly different to produce two orthogonal degenerative modal signals, but there are many ways to produce degenerative modal signals in a cavity. Other approaches that are seen in GNSS patch antennas are adding or subtracting tabs to a square conductive (top) patch, cutting the corner off the dielectric substrate, or any other technique to disrupt the perfectly symmetrical cavity structure of the patch to produce the two degenerative modal signals that have the same amplitude and are 90° out of phase at the feed location.

While the single-fed RHCP patch is a convenient design, the AR is usually not as good as a dual probe-fed RHCP antenna due to the asymmetry of the patch. In some implementations, the feed can be slightly adjusted off the diagonal to help match the impedance of the antenna and to help improve the AR.

Figure 5.9 is a photograph of a typical commercially available, low-cost active single-frequency RHCP GPS antenna that is probe fed. This patch is square (not nearly square) with the corner of the dielectric substrate cut to produce the asymmetry and two degenerative modal signals within the patch. The probe feed location is slightly off of the diagonal to help provide a good impedance match and to obtain a good AR. The probe is capacitively coupled to the patch; see small thin dielectric ring around the soldered probe in Fig. 5.9. The photo on the right in Fig. 5.9 illustrates the back side of the antenna that includes the associated RF bandpass filters, amplifiers, and voltage bias circuit, on the bottom side of the patch ground plane. As can be seen from both photos, the bottom RF components are enclosed within an "RF can" when the bottom circuit board is screwed into the metal body base. A plastic radome covers the top of the antenna (not shown).

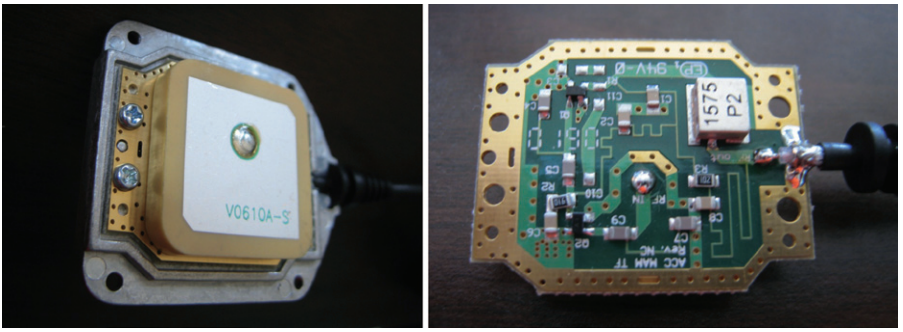


Fig. 5.9 Photograph of a low-cost active single-frequency probe-fed RHCP GPS antenna (radome not shown) (courtesy of u-blox).

5.4.2.5 Dual Probe-Fed, RHCP, Multifrequency GNSS Patch Antenna

Multiple frequencies can be supported in a multifrequency patch antenna design by stacking (i.e., layering) the patch elements on top of each other vertically. Figure 5.10 illustrates a dual-frequency probe-fed LP patch design. For the dual-frequency patch illustrated in Fig. 5.10, the input impedance of the first patch (i.e., lower patch) off resonance will have a very low real part (about 0Ω) at the desired resonance frequency of the second patch (i.e., top patch). This low real part of the input impedance at the feed port (at the operating frequency of the upper patch) will effectively add in series to the input impedance, at the feed port, for the input impedance at the lower patch frequency [18]. This is generally true provided each patch operates outside the Q_t of each others bandwidth, and that they do not operate at frequency harmonics of the each other. Because each patch has a relatively narrow bandwidth, when the frequency bands to be supported are fairly far apart (e.g., greater than at least 10%) [18], then there is no prohibitive mutual coupling between the two patch elements. Now this isolation is not exclusive, and in reality, each patch does affect each other, but multiple resonance frequencies can be supported with proper tuning.

In Fig. 5.10, the feed comes up from the bottom, through the ground plane on the bottom of the lower substrate, and is typically capacitively coupled to the lower patch, and then either physically connected to the upper patch element or, again, capacitively coupled to the top patch [21, 22]. The return/ground of the feed is attached to the bottom of the ground plane.

The various design techniques for probe-fed LP, single-frequency patch antennas can be applied to multifrequency patch antennas, with the capacitive coupling technique illustrated in Fig. 5.10. As presented earlier for the probe-fed LP, single-frequency patch antennas, the impedance of the antenna can be matched to the transmission line/connector by moving the feed location from the edge toward the center of the patch. Keeping in mind that the length of each patch is determined by the desired resonance frequency, careful attention

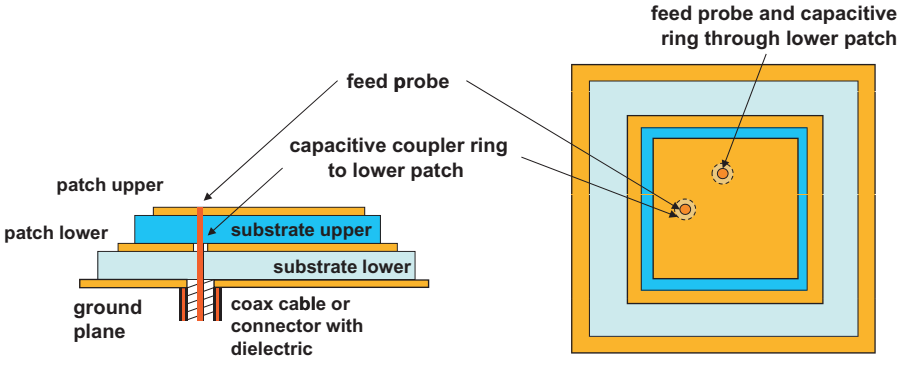


Fig. 5.10 Dual probe-fed, RHCP, multifrequency GNSS patch antenna.

must be given to the final feed location to obtain good impedance matches, at all of the desired resonance frequencies. Often this is a compromise between the impedance match obtained at the multiple frequencies to be supported by the antenna. With the dual probe-fed RHCP approach to support multiple frequencies, each of the LP probe ports can then be combined with a hybrid (i.e., 90°) power combiner to produce an RHCP output signal.

5.4.3 Survey-Grade/Reference GNSS Antennas

This section is devoted to what can be categorized a high-quality, survey-grade, geodetic, and/or reference station GNSS antennas. There are a significant number of these types of antennas used for various high-performance GNSS applications, including geodetic survey, high-precision farming/construction/machine control, and fixed GNSS reference station. Almost all are dual or multifrequency to support removal of the error introduced due to the ionosphere. These types of applications also place a premium on performance including multipath mitigation to enable high precision. Most often, these types of antennas are larger than simple patch antennas and are more costly, even though they may incorporate patch antenna technologies into their design. Once again, there are a wide variety of designs in the marketplace, and this section will present a few of the more common configurations.

5.4.3.1 Choke Ring-Based GNSS Antennas The choke ring-based antenna was originally introduced to the GPS community by the Jet Proposal Laboratory (JPL) and was used with a Dorne & Margolin dipole-based GPS antenna element. The choke ring is essentially a shaped ground plane made of conductive material forming a series of concentric circular troughs for the purpose of mitigating multipath. Figure 5.11 illustrates a common configuration of a typical (i.e., 2D) choke ring utilized in GNSS applications.

Typical configurations include three to four choke rings, of depth slightly greater than a quarter wavelength and width slightly greater than an eight

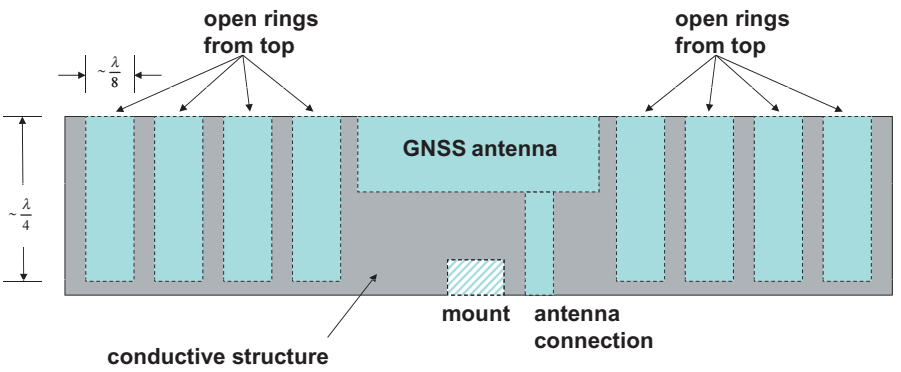


Fig. 5.11 Typical choke ring-based GNSS antenna configuration.

wavelength [24]. The geometric variation of the ground plane provides a reduction in gain for lower elevation angles to the GNSS SVs. Multipath and interference signals below the horizon (i.e., 0° elevation angle) are diffracted by the choke rings. The diffractions will induce secondary currents on the choke ring structure, and some current will find its way to the antenna elements but will be substantially reduced had the choke ring structure not been present.

The physical dimensions of the choke ring ground plane structure can be optimized for a single GNSS frequency, or designed to be a compromise in performance for multiple GNSS frequencies that are several hundreds of megahertz apart.

Various antenna designs can be used at the center of the choke ring as the radiating element. Initially, a domed (i.e., bubblelike) radome housing was commonly used to enclose a dipole-based GPS antenna. Today, multifrequency patch antennas are also commonly placed at the center of GNSS choke ring antennas.

Variations in the traditional choke ring structure have also been used to help mitigate multipath error, including a variation of depth [25, 26], use of a single choke ring [27], and a 3D choke ring structure [28]. Figure 5.12 illustrates a 3D choke ring structure [29]. The latest generation of 3D choke ring structures provide for a more uniform gain in the upper hemisphere and better suppression of the gain at negative elevation angles for multipath and interference mitigation.

5.4.3.2 Advanced Planner-Based GNSS Antennas Aside from choke ring-based antennas, there is another class of high-performance GNSS antennas that are mostly planar (i.e., microwave circuit board based) in design used in geodesy and reference station applications. While there are several antennas that can be categorized as such, information on three antennas,

Fig. 5.12 Photograph of a 3D choke ring [29] (courtesy of NovAtel).

the Roke geodetic, NovAtel Pinwheel, and Trimble Zephyr antenna, will be presented here.

The Roke geodetic-grade antenna is a broadband spiral-type antenna designed to support triple-frequency GNSS bands [30]. The antenna is a cavity backed spiral with a unique ground plane structure to help minimize multipath [31]. Figure 5.13 is a photograph of the spiral radiating element above the cavity backed structure (i.e., metal can). A cavity backed structure attempts to minimize the reflection off the bottom of the can with special absorbing material within, and what is left reflects in phase with the upward signal in the upper hemisphere. The radiation spiral elements are typically placed $\lambda/4$ above the inside bottom of the can structure. (The total travel distance of the signal inside the can will be 180° and, with a 180 phase shift caused by the normal incident reflection from the metallic can structure, will cause the reflected signal [at zenith] to be in-phase with the upward signal.)

The NovAtel Pinwheel antenna is a planar-type antenna design that operates in a broadband fashion with its performance characterized and specified for specific GNSS signal bands [32]. The pinwheel has 12 spiral radiation elements that are aperture coupled from a circular feed on the bottom, with enhancements for multipath mitigation [33, 34]. The radiation elements for one of the latest generation of the pinwheel antennas are shown in Fig. 5.14. The radiation characteristics for the triple-frequency GNSS pinwheel are illustrated in Fig. 5.15 in the L1, L2, and L5 frequency bands. The GNSS pinwheel antenna provides a reduction of gain at low elevation angles and significant gain suppression below the horizon to mitigate multipath.

Fig. 5.13 Spiral GNSS antenna (spiral and can only) [30] (courtesy of Roke).

Fig. 5.14 Enhanced pinwheel GNSS antenna (radome removed) plane [35] (courtesy of NovAtel).

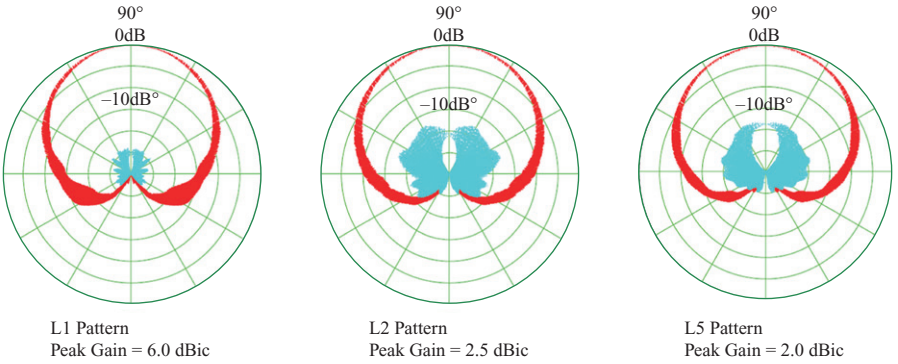


Fig. 5.15 Pinwheel radiation characteristics in elevation plane [35] (courtesy of NovAtel).

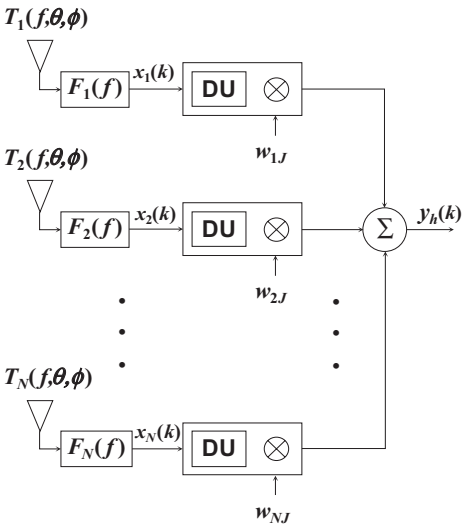
The Trimble Zephyr GNSS antenna is another planar-type GNSS antenna that implements advanced technologies to help mitigate multipath for mainly geodetic and reference ground station applications. The Zephyr antenna utilized a six-feed patch antenna design with a resistively tapered ground plane, where the surface resistance increases as the distance from the center of antenna increases to provide for enhanced performance and multipath mitigation [36].

5.5 PRINCIPLES OF ADAPTABLE PHASED-ARRAY ANTENNAS

Phased-array antennas implement multiple antenna elements where the physical orientation, phase, and/or amplitude can be controlled to obtain superior performance above that of a single antenna (i.e., a fixed reception pattern antenna [FRPA]). Adaptable phased-array antennas, often called smart antennas or controlled reception pattern antenna (CRPA) in the GNSS community, have the capability to control and change the net antenna radiation pattern characteristics as a result of sensing the environmental conditions that are presented to the GNSS antenna and/or receiver system. The sensing of the environmental conditions presented to the GNSS antenna and/or receiver system can include sensing the host platform attitude with a host inertial navigation system (INS) or integrated inertial measurement unit (IMU), decoding or receiving information pertaining to the location of the GNSS SVs to be received, measuring/estimating the composite signal power level within the entire band, and/or estimating the signal power levels of individual signals within the band. When the sensing and adjusting is done iteratively over time, the process is adaptive to account for the changing environmental conditions to optimize the GNSS signal measurements.

Various signal processing techniques are implemented in GNSS adaptable phased-array antennas. Space adaptive processing (SAP) generally refers to a technique where the spatial relationship between the antenna elements in the array is used to optimize the antenna/receiver performance. SAP can be implemented with a power minimization, maximizing the signal-to-noise ratio, or a digital beamforming approach. Space–time adaptive processing (STAP) techniques add filtering, typically implemented digitally (e.g., finite impulse response [FIR] filter) to each antenna reception path to help increase the degrees of freedom and bandwidth response of array. Space–frequency adaptive processing (SFAP) techniques perform signal processing in the frequency domain using digital filtering for interference/jamming signal mitigation. Digital beamforming techniques explicitly perform pattern shaping based on the desired signal and interference signals (or just the desired signal) locations and power levels.

A general block diagram of an antenna array that is used in an adaptive method for GNSS applications is shown in Fig. 5.16, including a description of the block functions and notation listed below the block diagram. For each of the N antenna elements in the array, the individual antenna elements will have a transfer function that will vary based on frequency and aspect angle, $T_n(f, \theta, \phi)$, which will feed front-end RF components such as RF amplifies, filters, and other components that can be represented by the transfer function $F_n(f, \theta, \phi)$. At the output of each of these antenna paths, complex weights (i.e., amplitude and phase modification) are applied to each signal path. (When STAP is not performed, the delay unit [DU] has no delay [i.e., $j = 0$].) After the complex antenna weights have been applied, the signals from each path can be combined, as represented in Eq. 5.11 for non-STAP processing, where the desired signals (s_d), interference (i), and noise (n) are received as the total signal (x),



$n = 1, 2, 3\ldots, N$ = antenna element index
N = max number of antenna elements
$T_N(f_q,\theta,\phi)$ = antenna element complex transfer function
$F_N(f_q)$ = RF front- end complex transfer function
$x_N(k) = s_d(k) + i(k) + n(k)$ = total signal, interference, and noise received
where
$s(k) = s_d(k) + i(k)$ = total signals received (desired plus interference)
$i(k)$ = total interference received
$n(k)$ = total noise received
k = time index
d = desired signal index = 1, 2, 3…, D
i = interference signal index = 0, 1, 2, 3…, I
z = total signal index = 1, 2, 3…, Z
D = max number of desired signals to be processed by the array
I = max number of interference signals to be processed by the array
$Z = D + I$ = max number of signals to be processed by the array
DU = delay unit
j = DU time index = 0, 1, 2,…, J
$w_{NJ}(k)$ = complex antenna array weights (to be calculated)
$y_h(k)$ = complex antenna array output
h = channel output number = 1, 2, 3,…, D

Fig. 5.16 General block diagram of a GNSS adaptive antenna array.

assuming the effects of $T_n(f, \theta, \phi)$ and $F_n(f, \theta, \phi)$ have been compensated for in an antenna calibration process. The output of the antenna array in Eq. 5.11 is shown as a channel output (h); this can be a single RF or IF port, or on a per SV basis depending upon the algorithm used to calculate the antenna weights, and the receiver configuration. The effects from the geometric position of the antenna elements are included in x, in the form of the antenna steering vector (i.e., geometric antenna array factor):

$$y_h(k) = \sum_{n=1}^{N} w_n(k) x_n(k), \quad \text{non-STAP (neglecting } J\text{).} \tag{5.11}$$

Without loss of generality, the signals in Eq. 5.11 are represented as discrete signals with time index k. Additionally, the details of where the functions of Fig. 5.16 are performed, and what methods are used to calculate the complex antenna weights will depend upon the performance requirements for the GNSS antenna and receiver systems, technologies employed, and user configuration constraints.

Historically, CRPA used with GPS receivers has been used by military users to help mitigate intentional jamming and unintentional interference [37]. These configurations typically involved the CRPA antenna, associated antenna electronics (AE), and a single-RF (or IF) input port on a GPS receiver [38].

Initial CRPA configuration performed an adaptive nulling technique to place nulls at the spatial location of jamming/interference sources to provide a single output to a GPS receiver. The objective of the adaptive nulling algorithm is to adjust the weighting of the antenna elements so that the power in the sum of the weighted signal output is minimized, subject to a constraint that prevents the minimized power from being below a certain value. Typically, the constraint is provided by fixing the weight of one of the antenna elements (i.e., a reference antenna element) and by allowing the other weights to be adjusted. In mathematical terms, the weights are adjusted to minimize the average power of the antenna output with respect to the reference element. Without loss of generality, it can be assumed that the constraint is applied by fixing the weight for the reference element path to unity. Thus, there are $N - 1$ "degrees of freedom" in adjusting the weights; as many as $N - 1$ nulls in the antenna spatial pattern can be generated.

The adaptive nulling power minimization technique described above works well to minimize the effects from the interference source but can have a negative effect on the desired GNSS signals to be tracked because the location of the desired signal locations is not considered in the antenna weight calculations. The most severe case is when the interference source is in the same direction as a desired signal direction. For certain high-performance applications, the distortion to the code and carrier phase measurements is significant, and more advanced techniques can be applied.

5.5.1 Digital Beamforming Adaptive Antenna Array Formulations

In addition to placing nulls in the directions of interference sources, directional beams can be pointed in the direction of the desired GNSS signals to help minimize the effects from the interference source on the GNSS code and carrier measurements. This type of technique is typically referred to as digital beamforming, and one popular algorithm is referred to as minimum variance distortionless response (MVDR) whereby the desired signal is passed undistorted, after application of the complex antenna array weights, while minimizing the output noise variance [39, 40, 41]. Theoretically, this type of technique will not distort the code and carrier measurement from the SV signal being processed, but practical limitation with the degrees of freedom in the antenna array, locations of interference sources with respect to the location of the desired SV signal, interference power level, and waveform type will limit the effectiveness of this distortionless processing of the desired signal. These types of digital beamforming algorithms are well suited for digitally based GNSS receiver signal processing (i.e., software-defined radio)-based architectures, where each GNSS signal can be processed, individually, in a digital receiver channel. Each of the N antenna path signal data will have the antenna weights applied, different for each SV to be processed, and will produce D digital channel outputs for code and carrier tracking. A separate set of weights is required for each beam pointed in the direction of each desired GNSS signal

to be tracked. The complex antenna weights are typically computed on the basis of antenna attitude information using either the host INS or integrated IMU, and GNSS satellite position data (e.g., ephemeris data), so that each beam points toward a satellite.

For example, the output of channel 1 of the antenna array output to be applied to the digital channel 1 of the digital GNSS receiver can be represented as in Eq. 5.12 [39, 40, 41]. For distortionless processing of the desired signal s_{d1}, the product of the geometric antenna steering vector (i.e., antenna array factor, pointed in the estimated direction of the desired signal) and the Hermitian (H) transpose of the complex antenna weights (yet to be determined) should be 1:

$$\mathbf{y}_1(k) = \underbrace{\mathbf{w}^H \mathbf{a}_1(\theta_1, \phi_1)}_{\text{want this to be 1}} s_{d1} + \mathbf{w}^H \mathbf{u}, \quad \text{for} \quad h = 1, \tag{5.12}$$

where

$\mathbf{a}_1(\theta_1, \phi_1)$ = the N-element geometric antenna array steering vector in the direction of (θ_1, ϕ_1) for the desired signal s_{d1}
$\mathbf{w}$ = is the complex $[N \times 1]$ antenna array weights (yet to be determined)
$[\,]^H$ = Hermetian transpose (i.e., complex conjugate transpose)
$\mathbf{u}$ = undesired signal vector (interference + noise).

The expected value of Eq. 5.12 will be s_{d1}, and the variance can be calculated as $\text{VAR}[\mathbf{y}_1] = \mathbf{w}^H \mathbf{R}_{uu} \mathbf{w}$, where the undesired signal correlation matrix is expressed as $\mathbf{R}_{uu} = \text{E}[\mathbf{u}\mathbf{u}^H]$. To optimally solve for the complex antenna weights for the MVDR process, the method of Lagrange can be used to define a cost function that is a linear combination of the variance of the output and the constraint that $\mathbf{w}^H \mathbf{a}_1(\theta_1, \phi_1) = 1$. Minimizing this cost function leads to the solution for the antenna weight shown in Eq. 5.13, for channel 1 [39, 40, 41]:

$$\mathbf{w}_{mv1} = \frac{\mathbf{R}_{uu}^{-1} \mathbf{a}_1(\theta_1, \phi_1)}{\mathbf{a}_1^H(\theta_1, \phi_1) \mathbf{R}_{uu}^{-1} \mathbf{a}_1(\theta_1, \phi_1)} \quad \text{(for a single signal per receiver channel)}, \tag{5.13}$$

where

$\mathbf{R}_{uu} = E[\mathbf{u}\mathbf{u}^H]$ = undesired signal correlation matrix
$E[\,]$ = expected value function.

The complex antenna weights expressed in Eq. 5.13 would be used to process the desired signal 1, that is, s_{d1}, and applied to digital receiver channel 1. This process places a beam in the direction of s_{d1}. A similar process would be completed for each of the other desired GNSS signals to be processed into each of the digital GNSS receiver channels.

Figure 5.17 illustrates a typical theoretical performance of a seven-element CRPA-type configuration array factor for a desired signal at an elevation angle of 80° and azimuth angle of 90°, where the view angle is at an elevation angle of 30° and azimuth angle of 10°. The azimuth scale represents the horizon and the upper hemisphere would represent the radiation characteristics pointed toward the SV (the lower half of the theoretical array factor is shown but in reality would be suppressed by the ground plane) [42].

In addition to digital beamforming to place a directional beam in the direction of the desired signal to be digitally processed, nulls can be placed in the direction of interference sources while minimizing the output variance of the desired signal. This can be accomplished by including an estimate of the interference signal direction in the antenna steering vector and including this in the undesired signal correlation matrix shown in Eq. 5.13. A similar process would be completed for each of the other desired GNSS signals to be processed into each of the digital GNSS receiver channels.

Using a reference element in the antenna array, the degrees of freedom available for interference mitigation will be $N - 1$, but the effectiveness for interference mitigation will be a function of the number of elements in the array, locations of interference sources with respect to the location of the

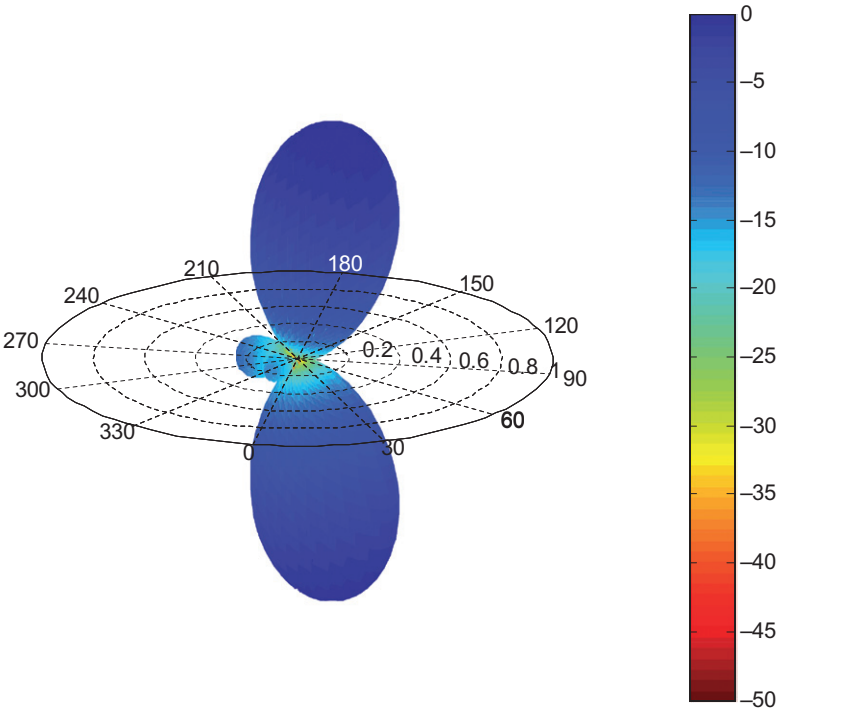


Fig. 5.17 Illustration of theoretical seven-element CRPA array factor (with no ground plane) [42].

desired SV signal, interference power levels, and waveform type. It has been shown that strong and broadband interference signals consume more that just one degree of freedom in the array processing [43]. Methods to increase the degrees of freedom are to increase the number of geometric elements [42] or to implement advanced digital signal processing with the same number of elements.

5.5.2 STAP

The STAP technique provides for better interference mitigation above the basic SAP processing technique when the geometric degrees of freedom in the antenna structure are consumed by increasing interference power and/or waveforms [43]. To help increase the interference mitigation performance, digital filters (i.e., FIR) can be added to each antenna path to help increase the bandwidth response of the array. In Fig. 5.16, the DU function is expanded from 0 to some integer number (i.e., J), and each of the J weights (i.e., $w_{nj}(j)$) are applied to each delayed version of the signal (i.e., $x_{nj}(j)$). The weighted outputs of each channel, $x_{nJ}(k)$, are then summed to produce the antenna array output in accordance with Eq. 5.14:

$$\begin{aligned} x_{nJ}(k) &= \sum_{j=1}^{J} w_{nj}(j)\, x_{nj}(j) \\ y_h(k) &= \sum_{n=1}^{N} x_{nJ}(k), \quad \text{with STAP (finite } J\text{).} \end{aligned} \tag{5.14}$$

5.5.3 SFAP

The interference reduction capability of SAP-only processing is best when the interfering signals are narrowband. Broadband jammers present a problem, because for a given set of antenna weights, the antenna spatial gain pattern varies with frequency. Thus, a set of weights optimal at one frequency will not be optimal at another frequency. SFAP solves this problem by dividing the frequency band into multiple narrow sub-bands, typically by using a fast Fourier transform. For each sub-band, a set of optimal weights is used to obtain a corresponding desired antenna spatial pattern for that sub-band. By combining the nullformed or beamformed signals from all the subbands, an optimal wideband antenna spatial pattern is obtained. Similar results can be obtained by using a bank of narrowband filters in the time domain, as presented previously [44].

5.5.4 Configurations of Adaptable Phased-Array Antennas

Various configurations of adaptable phased-array antenna have been used, mostly by military users for air and sea applications. The seven-element CRPA

(e.g., CRPA and CRPA-2) with AE (e.g., AE-1) has been integrated with single RF and IF input GPS receivers to form nulls in the location of interference sources. The AE does a significant amount of the processing to adaptively process the signals received from the CRPA antenna for null forming. A modernized version of the CRPA-2/AE-1 configuration is the GPS Antenna System-1 (GAS-1). The follow-on to the GAS-1 is the Advanced Digital Antenna Production (ADAP) system, which will provide for enhanced interference mitigation and beamforming capabilities [45].

One of the newest generations of GPS antenna/receiver systems to implement digital beamforming and nulling technology, with spatial temporal adaptive processing (STAP), is the GPS Spatial Temporal Anti-Jam Receiver (G-STAR). The G-STAR has the ability to handle narrowband and wideband interference sources with its STAP and to place nulls toward the interference sources while directing high gain beams in the directions of the desired SV signals [46].

Due to the physical size of the seven-element CRPA, efforts have gone into developing smaller adaptable phased-array antennas for various platforms, especially smaller vehicles. Small or compact CRPAs have emerged and several four-element CRPA-type configurations with a smaller footprint are available [47].

5.5.5 Relative Merits of Adaptable Phased-Array Antennas

- Adaptive nulling is much simpler and cheaper than beamforming since only one output emerges from the process, enabling use with a more traditional single-input GNSS receiver.
- A beamforming antenna produces one output for each beam, so a considerably more complex receiver is required to process each output independently; typically implemented in a digital receiver, capable of processing each channel independently.
- Beamforming can produce significant spatial gain in the direction of the GNSS satellites, while adaptive nulling makes no attempt to maximize gain in the desired directions.
- Jamming reduction with adaptive nulling antenna only techniques can be limited for strong or broadband interference waveforms; STAP- and SFAP-type techniques can be added to increase the array bandwidth response and to improve interference mitigation robustness.
- Beamformers tend to have high spatial gain in the direction of desired signals and lower gain in other directions. Multipath arriving from a low-gain direction is therefore attenuated relative to the desired signal.
- Because of the large physical extent of the antenna array, beamformers and nullers have the common problem of causing biases in signal delay caused by movement of the antenna phase center as a function of the weight values. In some cases, biases of 100° in carrier phase and 1 m in

code phase can occur. For high-precision systems, these errors can be significant and require compensation/calibration [48].

5.6 APPLICATION CALIBRATION/COMPENSATION CONSIDERATIONS

This section addresses issues regarding calibration of GNSS antennas, compensation for group and phase delays due to the antenna design, surroundings, or operational effects on the code and carrier phase measurements. Depending upon the end-user environmental and performance requirements, the level of attention to code and carrier phase measurement compensation may vary. While the GNSS antenna design requirement is to provide the GNSS signals to the GNSS receiver so that they can effectively be processed, the quality and fidelity will depend on the application, cost, size, and so on.

Generally, the larger an antenna is, and the closer things get to the radiating element of an antenna, the more variation there will be in antenna performance. At the antenna design level, items (ground planes, mounts, radomes, lightning rods, filters, power supply, etc.) in close proximity can have a significant effect on the performance. This is especially true for small consumer products that must conform to a compact form factor. Typically, other device components close to the antenna are modeled in a CEM along with the antenna so that performance predictions for the antenna "as installed in its operational environment" can be accomplished. This is also true for radome structures that will shift the operating frequency of the antenna downward. For example, a patch antenna with a plastic-type radome a couple of millimeters thick, placed a couple of millimeters above the radiating patch, will shift the operating frequency downward on the order of 4–6 MHz, which, of course, changes the amplitude and phase response of the antenna. This type of compensation must be done at the design stage of the antenna including its structure.

After the antenna has been produced, for high-performance and/or high-integrity applications, calibration/compensation of the code and/or carrier phase measurements may be needed with respect to an antenna reference point (ARP). The ARP is a physical point on the antenna that code and/or carrier phase measurements are referenced to. Additionally, depending upon application, it is also useful to establish an azimuth reference marker on the antenna to calibrate/compensate for possible azimuth variation of the code and/or carrier phase measurements. Most often, this is accomplished with a "north marker" tab for stationary applications, or aligning the antenna in a particular fixed direction with respect to the body frame of a dynamically moving vehicle.

For survey-grade/reference GNSS antennas used in geodetic and general-purpose reference station-type applications, the carrier phase and/or code phase compensation is often a very important item that must be addressed

to enable high-accuracy kinematic processing with the carrier phase or for high-integrity applications. For multifrequency antennas, the U.S. National Oceanic and Atmospheric Administration, National Geodetic Service (NGS) provides individual absolute calibration for a wide variety of multifrequency GNSS antennas [49]. Additionally, commercial companies have provided GNSS user equipment and GNSS SV antenna calibration services [50].

For high-performance GNSS antennas used in GNSS reference station-type applications, the code phase compensation is often a very important item that must be addressed to enable high accuracy and high integrity. Reference station applications used to support precision landing systems must limit the multipath caused by the ground and are often large antennas. Larger antennas will typically have more code and carrier phase variation versus the spatial coordinates to the satellite that are being tracked. For these types of applications, antenna modeling [51], anechoic chamber testing [52], and field testing [53] can be used to build confidence in the code and/or carrier phase compensation/calibration process employed.

The adaptable phased-array antennas presented earlier have very unique antenna calibration requirements. First, by their nature of being in an antenna array, the amplitude and phase response of each individual antenna element, $T_n(f, \theta, \phi)$ in Fig. 5.16, will be different, based on their location in the array. Additionally, these transfer functions may, and will likely change, based on the "as installed" (i.e., *in situ*) configuration of the antenna array. Furthermore, when interference is present, the complex amplitude weight algorithm will compute and apply the antenna weights to each signal path in accordance with the technique used. These techniques, even for an MVDR algorithm, can produce significant amplitude and phase distortion depending upon the degrees of freedom in the antenna array, locations of interference sources with respect to the location of the desired SV signal, interference power level, and waveform type. Significant effort has gone into providing premission and online calibration for these types of applications [48].

For other applications, such as wraparound microstrip GNSS antennas on spinning missiles, projectiles, or launch vehicles, measurement compensation is important. Since the code phase is unambiguous, provided the signal antenna gain pattern presented to the SV is fairly uniform, and the code tracking loop is able to maintain reasonable C/N_o, then the code measurements should not see significant error. However, significant attention must be given to the carrier phase measurement if it is continuously tracked and used. Since the carrier phase measurement is ambiguous, with a continuously tracked carrier phase measurement on-board a spinning vehicle, the carrier phase measurement will "wind up" depending upon the vehicle spin rate and aspect angle to the SV. An IMU or INS can be used to compensate for this carrier phase windup to "unwind" the carrier phase measurement, based upon the spin rate and attitude with respect to the aspect angle to the SV signal being tracked.

PROBLEMS

5.1 A passive GNSS antenna with a measured input impedance of $Z_A = 45 - j15\,\Omega$ is connected to a small transmission line (RF microstrip line) with a characteristic impedance of $Z_o = 52 + j10\,\Omega$. At this interface connection, what is the

(a) refection coefficient,

(b) standing wave ratio,

(c) mismatch efficiency, and

(d) return loss?

5.2 Consider an antenna coordinate system and crossed-dipole antenna configuration presented in Fig. 5.2, with an incident GNSS signal for the form $E_{GNSS}(z, t) = \cos(\omega_c t)a_x - \sin(\omega_c t)a_y$. If the input ports of the 90° power hybrid combiner are reversed (i.e., dipole port 1 is delayed by 90°), calculate the output signal $s(t)$.

5.3 Design an edge probe-fed square patch antenna to operate at the GPS L1 frequency with a Rogers RT/duroid® 6002 dielectric substrate that has a relative permittivity of 2.9 and a height of 3.0 mm. The probe feed has a diameter of 0.7 mm and is to be placed at the center of the feed side. Perform the following:

(a) Sketch the patch antenna from the side and top; label all important items.

(b) What is the physical size (i.e., length and width) of the top radiation patch elements?

(c) What is the input impedance for the antenna?

(d) What polarization will the patch antenna be sensitive to?

(e) Plot the real part of the input impedance if the feed is moved from the edge to the center, and then to the far edge, along the center line of the patch.

(f) At what distance from the edge of the patch would the real part of the input impedance be 50 Ω?

5.4 Design a single, probe-fed nearly square patch antenna to receive the RHCP GPS L1. The probe feed is to be placed along the diagonal of the patch. The performance requirement is to have an AR no more than 2 (at zenith). The dielectric substrate has a relative permittivity of 9.8 and a height (i.e., vertical thickness) of 6 mm. The antenna is expected to have a total Q of 10. Perform the following:

(a) Calculate the physical size (i.e., length and width) of the top radiation patch elements.

(b) What are the two degenerative modal signal frequencies for your design?

(c) Sketch the patch antenna from the top view; label all important items.

REFERENCES

[1] IEEE, *IEEE Standard Definitions of Terms for Radio Wave Propagation*, IEEE Std 211-1997(R2003), IEEE, Inc., New York, 1997.

[2] IEEE, *IEEE Standard Definitions of Terms for Antennas*, IEEE Std 145-1993(R2004), IEEE, Inc., New York, 1983.

[3] C. A. Balanis, *Antenna Theory, Analysis and Design*, 3rd ed. John Wiley & Sons, Inc., Hoboken, NJ.

[4] A. J. Van Dierendonck, P. Fenton, and T. Ford, "Theory and Performance of Narrow Correlator Spacing in A GPS Receiver," *ION Navigation Journal* **39**(3), pp. 265–284 (1992).

[5] T. Pratt, C. Bostian, and J. Allnutt, *Satellite Communications*, 2nd ed. John-Wiley & Sons, New York, 2003.

[6] RTCA, Inc., MOPS for GNSS Active Antenna Equipment in L1 Frequency Band, SC-159, DO-301, December 13, 2006.

[7] FEKO, FEKO Comprehensive Electromagnetic Solutions, 2012, available at: http://www.feko.info/, visited May 26, 2012.

[8] IE3D, Electromagnetic Simulation Solutions, 2012, available at: http://www.mentor.com/electromagnetic-simulation/, visited May 26, 2012.

[9] WIPL-D, Electromagnetic Modeling of Composite Metallic and Dielectric Structures, 2012, available at: http://www.wipl-d.com/, visited May 26, 2012.

[10] HFSS, ANSYS Electromagnetics, 2012, available at: http://www.ansys.com/Products/Simulation+Technology/Electromagnetics, visited May 26, 2012.

[11] XFdtd, REMCOM, XSdtd®, 2012, available at: http://www.remcom.com/xf7, visited May 26, 2012.

[12] G. Deschmps and W. Sichak, "Microstrip Microwave Antennas," *Proceedings of the 3rd Symposium on USAF Antenna Research and Development Program*, October 18–22, 1953.

[13] Sensor Systems, GPS S67-1575-135 GPS Antenna, http://www.sensorantennas.com/antenna_pdf/GPS/S67-1575-135%20Data%20Sheet.pdf, visited November 17, 2012.

[14] ARINC, GNSS SENSOR ARINC CHARACTERISTIC 743A-4, published December 27, 2001.

[15] Rogers, Advanced Circuit Materials. 2012, available at: http://www.rogerscorp.com/acm/literature.aspx, visited May 24, 2012.

[16] Taconic, RF & Microwave Laminates, 2012, available at: http://www.taconic-add.com/en–index.php, visited May 24, 2012.

[17] D. R. Jackson and N. G. Alexopoulos, "Simple Approximate Formulas for Input Resistance, Bandwidth, and Efficiency of a Resonant Rectangular Patch," *IEEE Transactions on Antennas and Propagation* **3**, 407–410 (1991).

[18] R. C. Johnson, *Antenna Engineering Handbook*, Chapter 7 by R. E. Munson. McGraw Hill, New York, 1993.

[19] W. F. Richards, "Microstrip Antennas," Y. T. Lo and S. W. Lee (Eds.), *Antenna Handbook*. Van Nostrand Reinhold Co., New York, 1998.

[20] W. F. Richardson, J. R. Zinecker, R. D. Clark, and S. A. Long, "Experimental and Theoretical Investigation of the Inductance Associated with a Microstrip Antenna Feed," *Electromagnetics* **3**(3–4), 327–346 (1983).

[21] R. Garg, et al., *Microstrip Antenna Design Handbook*. Artech House, Boston, 2001.

[22] P. S. Hall, "Probe Compensation in Thick Microstrip Patches," *Electronics Letters* **23**, 606–607 (1987).

[23] W. F. Richardson, Y. T. Lo, and D. D. Harrison, "An Improved Theory of Microstrip Antennas with Applications," *IEEE Transactions on Antennas and Propagation* **AP-29**(1), 38–46 (1981).

[24] UNAVCO, Choke Ring Antenna Calibrations, 2012, available at: http://facility.unavco.org/kb/questions/311/Choke+Ring+Antenna+Calibrations, visited May 28, 2012.

[25] V. Filippov, D. Tatarnicov, J. Ashjaee, A. Astakhov, I. Sutiagin, "The First Dual-Depth Dual-Frequency Choke Ring," Frequency Choke Ring, *ION GPS*, 1998, pp. 1035–1040.

[26] Javad, Javad Choke Ring Theory, 2012, available at: http://www.javad.com/jns/index.html?/jns/technology/Choke%20Ring%20Theory.html, visited May 28, 2012.

[27] D. B. Thornberg, D. S. Thornberg, M. F. DiBenedetto, M. S. Braasch, F. van Graas, and C. Bartone, "LAAS Integrated Multipath-Limiting Antenna," *ION Navigation Journal* **50**(2), pp. 117–130 (2003).

[28] W. Kunysz, "A Three Dimensional Choke Ring Ground Plane Antenna," *ION GPS/GNSS 2003*, Portland, OR, September 9–12, 2003, pp. 1883–1888.

[29] NovAtel, Antennas GNSS-705, 2012, available at: http://www.novatel.com/assets/Documents/Papers/GNSS-750.pdf, visited June 4, 2012.

[30] Roke, Roke Triple GNSS Geodetic-Grade Antenna, 2012, available at: http://www.roke.co.uk/resources/datasheets/042-GNSS-Antenna.pdf, visited June 7, 2012.

[31] R. Granger and S. Simpson, "An Analysis of Multipath Mitigation Techniques Suitable for Geodetic Antennas," *ION GNSS*, Savannah, GA, September 16–19, 2008.

[32] NovAtel, Antennas, GPS-703-GGG Data Sheet, 703-GGG, 2012, available at: http://www.novatel.com/assets/Documents/Papers/GPS-703-GGG.pdf, visited June 4, 2012.

[33] W. Kunysz, "A Novel GPS Survey Antenna," *ION NTM 2000*, Anaheim, CA, January 26–28, 2000, pp. 698–705.

[34] W. Kunysz, "High Performance GPS Pinwheel Antenna," *ION GPS 2000*, Salt Lake City, UT, September 19–22, 2000, pp. 2506–2511.

[35] NovAtel, GPS-704X Antenna Design and Performance, 2012, available at: http://www.novatel.com/assets/Documents/Papers/GPS-704xWhitePaper.pdf, visited June 4, 2012.

[36] E. Krantz, S. Riley, and P. Large, "The Design and Performance of the Zephyr Geodetic Antenna," *ION GPS 2001*, Salt Lake City, UT, September 11–14, 2001, pp. 1942–1951.

[37] NAVSTAR, Navstar GPS User Equipment Introduction, September 1996, available at: http://www.navcen.uscg.gov/pubs/gps/gpsuser/gpsuser.pdf, visited September 12, 2010.

[38] NSSRM, National Security Space Road Map, 2012, available at: http://www.fas.org/spp/military/program/nssrm/initiatives/crpa.htm, visited May 28, 2012.

[39] R. T. Compton, *Adaptive Antennas*. Prentice Hall, Englewood Cliffs, NJ, 1988.

[40] L. C. Godara, *Smart Antennas*. CRC Press, Boca Raton, Florida, 2004.

[41] F. Gross, *Smart Antennas for Wireless Communications*. McGraw Hill, New York, 2005.

[42] C. G. Bartone and T. Stansell, "A Multi-Circular Ring CRPA for Robust GNSS Performance in an Interference and Multipath Environment," *ION GNSS 2011*, Portland, OR, September 20–23, 2011.

[43] T. D. Moore and I. J. Gupta, "The Effect of Interference Power and Bandwidth in Space-Time Adaptive Processing," *Institute of Navigation, 59th Annual Meeting*, June 23–25, 2003.

[44] I. J. Gupta and T. D. Moore, "Space-Frequency Adaptive Processing (SFAP) for Interference Suppression in GPS Receivers," *ION NTM*, 2001.

[45] USAF, Advanced Digital Antenna Production System. U.S. Air Force, Los Angeles, 2012, available at: http://www.losangeles.af.mil/library/factsheets/factsheet.asp?id=18668, visited May 31, 2012.

[46] International On-line Defense Magazine, Defense Update, GPS Anti Jamming Techniques, 2012, available at: http://defense-update.com/products/g/GPS-STAP.htm, visited May 31, 2012.

[47] ITT, DM N100-3 Series GPS Antenna, ITT Exelis, Antenna Products and Technologies, 2012, available at: http://www.exelisinc.com/capabilities/Antennas/Documents/N100-3_Series.pdf, visited May 31, 2012.

[48] A. O'Brien, *Adaptive Antenna Arrays for Precision GNSS Receivers*, PhD Dissertation, Ohio State University, 2009, available at: http://etd.ohiolink.edu/view.cgi?acc_num=osu1259170076, visited May 30, 2012.

[49] NGS, National Oceanic and Atmospheric Administration, National Geodetic Service, Antenna Calibrations, 2012, available at: http://www.ngs.noaa.gov/ANTCAL/, visited May 30, 2012.

[50] Geo++, SMART in Positioning, 2012, available at: http://www.geopp.de/, visited May 30, 2012.

[51] D. Aloi, "Analysis of LAAS Integrated Multipath Limiting Antennas Using High-Fidelity Electromagnetic Models," *ION GNSS*, 2004.

[52] F. van Graas, C. Bartone, and T. Arthur, "GPS Antenna Phase and Group Delay Corrections," *ION NTM*, 2004.

[53] A. R. Lopez, "LAAS/GBAS Ground Reference Antenna With Enhanced Mitigation of Ground Multipath," *ION NTM*, 2008.

6

GNSS RECEIVER DESIGN AND ANALYSIS

6.1 RECEIVER DESIGN CHOICES

6.1.1 Global Navigation Satellite System (GNSS) Application to be Supported

One of the most important factors in GNSS receiver design is determining what application the receiver is going to be used in. An aviation receiver used for safety of live applications has very different requirements compared to a geodetic surveying receiver or a low-cost consumer-grade receiver used in a mobile cell phone. Many of the receiver design characteristics will be very different, but some characteristics will functionally be the same.

The intended application often drives the requirement to use a single GNSS (e.g., Global Positioning System [GPS] only) or a multiconstellation GNSS. For example, in low-cost consumer applications, a GPS L1 only receiver may be favored. For a more robust consumer application, a single-frequency but multiconstellation approach may provide better performance, for example, L1-only GPS and Global Orbiting Navigation Satellite System (GLONASS) and/or Galileo, and/or Compass/BeiDou. For a high-quality geodetic application where accuracy is paramount, a multifrequency approach will be favored for high-quality code and carrier phase measurements (e.g., L1 and L2 GPS), and multiconstellation support may be of secondary consideration. Aviation applications warrant either augmented single-frequency GNSS services or

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

dual-frequency GNSS support such as L1 GPS with Wide-Area Augmentation System (WAAS), GPS L1 and L5, or Galileo E1 and E5 support, where accuracy, continuity, integrity, and availability are key performance requirements.

6.1.2 Single or Multifrequency Support

Single-frequency GNSS support is often favored when the design constraints for multifrequency support in the GNSS receiver become too prohibitive for the application. This can include the size, cost, or power consumption needed to support a multifrequency GNSS receiver. If the performance requirements placed on the GNSS receiver can be met with a single-frequency GNSS system, then that is almost always favored. Some aviation applications utilize the single-frequency GNSS with augmentation, such as GPS with SBAS to satisfy ionosphere corrections and integrity requirements.

Multifrequency support in a GNSS receiver is most often driven by the requirement for enhanced accuracy and/or integrity for the intended applications. While there are some advantages for triple-frequency utilization in carrier phase applications, most pseudorange-based GNSS receiver applications benefit from dual-frequency support for ionosphere error measurement correction and/or frequency band protection utilization. Dual-frequency support in the GNSS receiver can also provide increased integrity either through removal of the dominate ionosphere bias error and/or provided integrity data (e.g., Galileo E5b).

6.1.2.1 Dual-Frequency Ionosphere Correction Because the error caused by the ionosphere is largely inversely proportional to the square of frequency, it can be calculated in a dual-frequency GNSS receiver by comparing the pseudorange measurements obtained on two frequencies. Using two GNSS pseudorange measurements, generally the further away these two frequencies are, the better the ionosphere error prediction will be. As will be presented in Chapter 7, the ionosphere error can be predicted by scaling and subtracting two pseudorange measurements. The predicted ionosphere error can then be subtracted from the measured pseudoranges. This processes removes the dominate ionosphere bias error but does increase the noise. This increased noise can then be averaged/smoothed. With the dominate bias removed, the smoothing time can increase substantially over any single-frequency smoothing approach [1].

6.1.2.2 Improved Carrier Phase Ambiguity Resolution in High-Accuracy Differential Positioning High-precision receivers, such as those used in surveying, use carrier phase measurements to obtain very precise range estimations. However, the periodic nature of the carrier makes the measurements highly ambiguous. Therefore, the solution of the positioning equations yields a grid of possible positions separated by a finite number of distances depending on code and carrier combinations, carrier phase ambiguity initialization, geometry, and measurement error. Removal of the ambiguity is accomplished by using additional information in the form of code pseudorange measure-

ments, changes in satellite geometry, or the use of more satellites. In general, ambiguity resolution becomes less difficult as the effective frequency of the carrier decreases. For example, by using both the GPS L1 and L2 carriers, a virtual carrier frequency of L1 − L2 = 1575.42 − 1227.60 = 347.82 MHz can be obtained, which has a wavelength of about 86 cm as compared to the 19-cm wavelength of the L1 carrier; this particular combination is often referred to as the *wide-lane* combination. Ambiguity resolution can therefore be made faster and more reliable by using this difference frequency, albeit with slightly less accuracy than with the L1-only solution. Various dual-frequency code and carrier combinations are possible [2] as well as triple-frequency techniques [3].

6.1.3 Number of Channels

GPS receivers must observe and measure GNSS navigation signals from at least four satellites to obtain three-dimensional position, velocity, and user clock error estimates. If the user altitude is known with sufficient accuracy, three satellites will suffice. There are several choices as to how the signal observations from a multiplicity of satellites can be implemented. Today, almost all GNSS receivers are considered *all-in-view receivers* that have enough channels to receive all desired satellites that are visible for a particular GNSS. For a single-frequency, single GNSS application, in most cases, 12 or fewer useful satellites are visible at any given time; for this reason, modern receivers typically have approximately 12 channels, with perhaps several channels being used for acquisition of new satellites or noise calculations whereby the remainder are for tracking. Use of more than the minimum of four satellites will materially improve the accuracy of the user solution by using an overdetermined solution.

For dual-frequency single GNSS applications, a 24-channel receiver would be considered an all-in-view receiver (e.g., GPS L1 and L2).

Additional receiver channels, on the order of 50 or so, provide added benefit to support either advanced measurement processing, such as multipath mitigation [4] or for multiconstellation GNSS support.

As semiconductor technology has advanced, so has the ability to place thousands of digital correlator channels on a single semiconductor device. Receiver architectures have been developed that advertise tens of thousands of digital correlator channels within a single device to rapidly search a multitude of carrier and code phase offsets simultaneously [5]. These types of GNSS correlator engines can be combined with microprocessors to provide a host-based GNSS user solution.

6.1.4 Code Selections

The signal code formats needed to support the planned GNSS service need to be supported by the GNSS receiver. The code rates and format will have an impact on the signal bandwidths, processor speed, memory, and eventual power consumption needed. Single, lower-rate codes, such as the GPS C/A-

code, can most easily be supported, whereas the Galileo E5a/b signal will demand the most bandwidth and processing. Codes that have error correction, authentication and/or encryption codes on them will require additional functionality to be processed within the GNSS receiver.

Commercial receivers can recover the L2 carrier without knowledge of the code modulation simply by squaring the received signal waveform or by taking its absolute value. More advanced squaring techniques take advantage of the underlying P(Y) code periodicity to obtain *semi-codeless* tracking to produce pseudorange measurements on the GPS L2 frequency (without use of the L2C code) [6]. Because the a priori signal-to-noise ratio (SNR) is so small, the SNR of the recovered carrier will be reduced by as much as 33 dB because the squaring of the signal greatly increases the noise power relative to that of the signal. However, the squared signal has an extremely small bandwidth so that narrowband filtering can make up the difference to produce a pseudorange estimate on the L2 frequency [6].

6.1.5 Differential Capability

Differential GNSS (DGNSS) is a powerful technique for improving the performance of a GNSS user solution. The performance increase can be realized in terms of accuracy, integrity, or availability for the particular application. This concept involves the use of not only the user's receiver (sometimes called *the remote or roving* unit) but also typically a *reference or monitor receiver*, and a *supporting data delivery method*. The complete treatment of DGNSS will be presented in Chapter 8. There are many ways to implement DGNSS and the implementation methods within the GNSS receiver can be just as varied.

DGNSS removes common systematic errors common to the user and monitor receiver. In a network-based DGNSS, there may be many sources of measurements or corrections. This chapter will focus on the corrections provided by or applied to a single GNSS receiver. The major sources of errors common to the reference and remote receivers, which can be removed (or mostly removed) by differential operation, are the following:

1. *Ionosphere Delays.* Ionosphere signal propagation group delay, which is discussed further in Chapter 8, can be up to about 80 m during the day to 3–6 m at night. Receivers that utilize dual frequencies can largely remove these bias errors by applying the inverse square law dependence of delay on frequency. DGNSS will significantly remove this error contribution for single-frequency GNSS users.
2. *Troposphere Delays.* These delays, which occur in the lower atmosphere, are usually smaller than ionosphere errors and typically are in the 1- to 3-m range for higher-elevation satellites but can be significantly larger at low satellite elevation angles (e.g., up to about 40 m). The troposphere errors are difficult to measure directly with GNSS receivers and are often

mitigated through a model for non-DGNSS users. DGNSS will significantly remove the troposphere error contribution for all GNSS users.

3. *Ephemeris Errors.* Ephemeris orbit errors, which are the difference between the actual satellite location and the location predicted by satellite ephemeris orbital data, are typically less than 2 m and will undoubtedly become smaller as satellite tracking technology improves. DGNSS will significantly remove the orbit error contribution for all GNSS users.
4. *Satellite Clock Errors.* These errors are the difference between the actual satellite clock time and the predicted satellite clock time, after applying the satellite clock error predictions from the satellite data. DGNSS will significantly remove the satellite transmitter clock error contribution for all GNSS users.

Differential operation can almost completely remove satellite clock and orbit ephemeris errors. For these quantities, the quality of correction has little dependence on the separation of the reference and roving receivers. The degree of correction that can be achieved for ionosphere and troposphere delays is excellent when the two receivers are in close proximity, such that the error terms are the same (i.e., do not decorrelate), for example, up to 20 km or so. At larger separations, the ionosphere and troposphere propagation delays to the receivers become less correlated, and residual errors after correction are correspondingly larger. Nonetheless, substantial corrections can often be made with receiver separations as large as 100–200 km. The amount of error mitigation needs to be compared to the performance requirements for the DGNSS applications. DGNSS is ineffective against noncommon errors such as multipath and receiver noise because these errors are strictly local to each of the receivers.

6.1.5.1 Corrections Formats In the broadest sense, there are several ways that differential corrections can be made and formatted for use. In a *solution-domain* approach, the reference station computes the position error that results from pseudorange measurements to a set of satellites, and this is applied as a correction to the user's computed position; a significant drawback to the solution-domain approach is that the user and reference station must use exactly the same set of satellites if the position correction is to be valid. Thus, the position domain DGNSS approach is not popular.

In the *measurement domain*, corrections are determined for pseudorange measurements to each satellite in view of the reference receiver, and the user simply applies the corrections corresponding to the satellites the roving receiver is tracking. Reference to a "lumped correction" means all of the error corrections are together in a single correction. Another method, which is still measurement based, decomposes error terms into individual terms (i.e., one for the ionosphere and one for orbit errors). Ground-based augmentation system (GBAS) and precise point positioning (PPP) DGNSS solution approaches tend to favor decomposition of error source terms.

In the majority of cases, it is important that corrections be applied as soon as the user has enough measurements to obtain a user solution, and for the corresponding ephemeris data set. Issuance of data (IOD) parameters can be used to ensure the correction data "matches" up with the correct basis in which the corrections were formed. When the user needs to know its corrected position in real time, current corrections can be transmitted from the reference receiver to the user via a data delivery method that may include a terrestrial or satellite link. The users can then receive, verify, and use the corrections in the user solution calculations. This capability requires a user receiver input port for receiving and using differential correction messages. While a user unique format could be used, standardized formats of these messages have been established by the Radio Technical Commission for Maritime Service (RTCM) Special Committee 104 (SC-104). Various versions of the RTCM SC-104 standard have been used over the years. Earlier versions tended to concentrate on robust pseudorange correction formats (e.g., Version 2.3), while later versions have emphasized carrier phase corrections (e.g., Version 3.0) and real-time delivery methods such as Networked Transport of RTCM via Internet Protocol (Ntrip) [7].

In some applications, such as surveying or non-real-time research truth reference systems, it may not be necessary to obtain differentially corrected position solutions in real time. In these applications, it is common practice to obtain corrected positions at a later time by bringing together recorded data from both receivers.

6.1.6 Aiding Inputs

Although various GNSS receivers can operate in a stand-alone system, navigation accuracy, coverage, and/or system availability can be materially improved if additional information supplements the GNSS receiver to aid in acquiring and/or tracking the received GNSS signals. Basic GNSS receiver aiding sources include the following:

1. *Inertial Navigation System (INS) Aiding.* Although GNSS navigation is potentially very accurate, periods of poor signal availability, jamming, and high-dynamics platform operations often limit its capability. INSs are relatively immune to these situations and thus offer powerful leverage in performance under these conditions. On the other hand, the fundamental limitation of INS long-term drift is overcome by the inherent calibration capability provided by a GNSS. Incorporation of INS measurements is readily achieved through Kalman filtering.
2. *Aiding with Additional Navigation Inputs.* Kalman filtering can also use additional measurement data from navigation systems, such as vehicular wheel sensors and magnetic compasses, to improve navigation accuracy and reliability.
3. *Altimeter Aiding.* A fundamental property of GNSS satellite geometry typically causes the greatest error in the user solution to be in the vertical

direction. Vertical error can be reduced by inputs from an absolute barometric pressure altitude sensor; however, a more common integration method is to relate the vertical solution height to a height reference, such as the ground using a barometric, radar, or laser altimeter sensor.

4. *User Clock Aiding.* An external clock with high stability and accuracy can be used by the user equipment to improve the user solution performance, but often only practical for stationary reference or time reference receiver applications. It can be continuously calibrated when enough satellite signals are available to obtain precise GPS time. During periods of poor satellite visibility, it can be used to reduce the number of satellites needed for user solution determination.
5. *Assisted GPS (A-GPS).* An assistance technique that has been applied to the indoor cellular market is A-GPS that provides a mobile station (MS), that is, a handset with assistance data from the cellular base station (BS). A-GPS data broadcast from a BS to an MS via the cellular network can provide the MS with the GPS broadcast navigation data and GPS system time to aid the receiver in initial GPS signal acquisition. These data allow the receiver to remove the space vehicle (SV) position, Doppler, 50-bps navigation data, and GPS system time uncertainly to expedite the initial signal acquisition. This is very useful in low SNR indoor environments [8].

6.2 RECEIVER ARCHITECTURE

Although there are many variations in GNSS receiver design, all receivers must perform certain basic functions. Figure 6.1 illustrates a generic block diagram illustrating these basic functions that are performed by GNSS receivers. The GNSS antenna was discussed in detail in the previous chapter. As depicted in Fig. 6.1, the GNSS antenna is illustrated as a passive device. We will now discuss the main GNSS receiver functions in further detail.

6.2.1 Radio Frequency (RF) Front End

The purpose of the receiver RF front end is to filter, amplify, and typically down-convert the incoming GNSS signal to an intermediate frequency (IF) signal that can be processed. Figure 6.2 illustrates a more detailed depiction of the RF front end and IF signal conditioning circuit. For high-quality RF front end circuits, an RF bandpass filter (BPF) is often placed directly after the passive antenna terminals. This RF passive BPF can be used to reduce out-of-band interference without degradation of the GPS signal waveform. Generally, the bandwidth of a BPF should be sufficient to pass, largely undistorted, the desired signal and have a sharp-cutoff out-of-band for signal rejection. However, the small ratio of passband width to carrier frequency makes the design of such filters unpractical for most GNSS receivers (and even undesirable at this stage of the receiver). Consequently, filters with wider skirts are

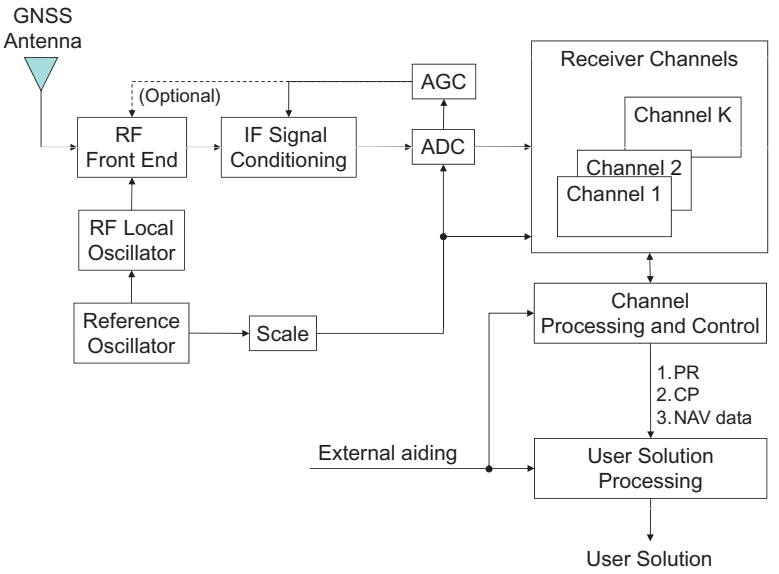


Fig. 6.1 Generic block diagram of GNSS receiver. PR: pseudorange; CP: carrier phase; NAV: navigation.

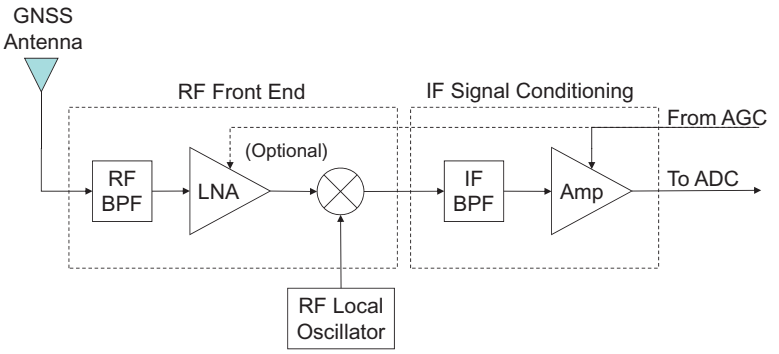


Fig. 6.2 Generic GNSS receiver RF front end and IF signal conditioning circuit.

commonly used as a first stage of filtering, which helps prevent front-end overloading by strong out-of-band interference sources, and the sharp-cutoff filters are used later after down-conversion to an IF. Furthermore, this RF BPF should be low loss (e.g., less than 1 dB or so) to keep the overall noise figure down and functions to provide two benefits: First, the RF BPF prior to the amplifier will attenuate the image frequency that can be directly down-converted to the IF frequency [9]. Second, the RF BPF will help attenuate any strong out-of-band interference from entering the low-noise amplifier (LNA) and will help prevent it from saturation.

As was pointed out earlier, the GNSS signal power available at the receiver antenna output terminals is extremely small and can easily be masked by interference from more powerful signals adjacent to the GNSS passband. To make the signal usable for digital processing at a later stage, RF amplification in the receiver front end typically provides as much as 35–55 dB of gain. The first-stage active amplifier (i.e., LNA) should have the highest gain and lowest noise to help minimize the overall receiver chain noise figure; however, not all the gain needs to be provided at the RF stage [10].

6.2.2 Frequency Down-Conversion and IF Amplification

After amplification in the GNSS receiver front end, the GNSS signal is typically converted to a lower IF, for further amplification and filtering. Down-conversion accomplishes several objectives:

1. The total amount of signal amplification needed by the receiver exceeds the amount that can be performed efficiently in the receiver RF front end at the GNSS carrier frequency. Excessive amplification at a particular stage in the receiver can result in parasitic feedback oscillation, which is difficult to control. In addition, since sharp-cutoff filters with a GPS signal bandwidth are not practical at the L-band, excessive front-end gain makes the end-stage amplifiers vulnerable to overloading by strong nearby out-of-band signals. By providing additional amplification at an IF different from the received signal frequency, a large amount of gain can be realized without the tendency toward oscillation.
2. The amplifiers used at either a first or second IF frequency are typically lower cost than an RF amplifier, which is especially true for the first RF LNA.
3. By converting the signal to a lower frequency, the signal bandwidth is unaffected, and the increased ratio of bandwidth to center frequency permits the design of sharp-cutoff bandpass filters. These filters can be placed ahead of the IF amplifiers to prevent saturation by strong out-of-band signals. The filtering is often by means of a high-order filter or surface acoustic wave (SAW) devices.
4. Conversion of the signal to a lower frequency makes the sampling of the signal required for digital processing much more feasible for small low-cost and low-power consumption applications.

Down-conversion is accomplished by multiplying the GNSS signal by a sinusoid called the *local oscillator signal* in a device called a *mixer*. The local oscillator frequency is either larger or smaller than the GNSS carrier frequency by an amount equal to the IF stage. Typically, the IF signal is selected to be the difference between the signal and local oscillator frequencies. The sum frequency components are also produced, but this is eliminated by a simple BPF

following the mixer. An incoming signal either above or below the local oscillator frequency by an amount equal to the IF will produce an IF signal, but only one of the two signals is desired. The other signal, called the image, can be eliminated by bandpass filtering of the desired signal prior to down-conversion. However, since the frequency separation of the desired and image signals is twice the IF, the filtering becomes difficult if a single down-conversion to a low IF is attempted. For this reason, down-conversion is often accomplished in more than one stage, with a relatively high first IF (30–100 MHz) to permit image rejection. Either a single stage of mixing (illustrated in Fig. 6.2) or double stage of mixing (i.e., super heterodyne) design can be utilized.

Whether it is single stage or multistage, down-conversion typically provides a final IF that is low enough to be digitally sampled at feasible sampling rates. In low-cost receivers, typical final IFs range from 4 to 20 MHz with bandwidths that have been filtered down to pass a substantial part of the GNSS code bandwidth (e.g., 2 MHz or so for a GPS C/A-code). This permits a relatively low digital sampling rate and at the same time keeps the lower edge of the signal spectrum well above 0 Hz to prevent spectral foldover. However, for adequate image rejection, either multistage down-conversion or a special single-stage image rejection mixer is required. In more advanced receivers there is a trend toward single conversion to a signal at a relatively high IF (30–100 MHz) because advances in technology permit sampling and digitizing even at these high frequencies.

Additionally, the theory of bandpass sampling has been applied to GNSS receiver designs. The theory of bandpass sampling allows for sampling frequency not to be a minimum of two times the carrier (or IF) frequency, but selected to be at least two times the bandwidth of the bandpass signal, with constraints based on the carrier frequency (or IF) and actual sampling frequency selected [11]. With regard to GNSS signal processing, the GNSS signal can be converted to an IF and then the signal is bandpass sampled to alias down the desired baseband signal band [12].

6.2.2.1 SNR An important aspect of receiver design is the calculation of signal quality as measured by the SNR in the receiver IF bandwidth. Typical IF bandwidths range from about 2 MHz in low-cost receivers to the full GPS signal bandwidth on the order of 20 MHz in high-end units, and the dominant type of noise is the thermal noise in the first RF amplifier stage of the receiver front end (or the antenna preamplifier if it is used). The noise power in this bandwidth is given by Eq. 6.1:

$$N = kT_eB, \tag{6.1}$$

where $k = 1.3806 \times 10^{-23}$ J/K, B is the bandwidth in hertz, and T_e is the effective noise temperature in degree Kelvin. The effective noise temperature is a function of sky noise, antenna noise temperature, line losses, receiver noise temperature, and ambient temperature. A typical effective noise temperature for a GNSS receiver is 513 K, resulting in a noise power of about −138.5 dBW in

a 2-MHz bandwidth and −128.5 dBW in a 20-MHz bandwidth. The SNR is defined as the ratio of signal power to noise power in the IF bandwidth or the difference of these powers when expressed in decibels. Assuming a nominal GNSS signal power level of −160.0 dBW, the SNR in a 20-MHz bandwidth is seen to be −160.0 − (−128.5) = −31.5 dB. About 90% of the C/A-code power lies in a 2-MHz bandwidth, so there is only about 0.5-dB loss in signal power. Consequently the SNR in a 2-MHz bandwidth is (−160.0 − 0.5) − (−138.5) = −21.5 dB. In either case, it is evident that the signal is completely masked by noise. Further processing to elevate the signal above the noise will be discussed subsequently.

6.2.3 Analog-to-Digital Conversion and Automatic Gain Control

In modern GNSS receivers, digital signal processing is used to track the GNSS signal, make pseudorange, Doppler, and carrier phase measurements, and demodulate the navigation data bit stream. For this purpose, the signal is sampled and digitized by an analog-to-digital converter (ADC) to be processed digitally. In most receivers, the final IF signal is sampled, but in some, the final IF signal is converted down to an analog baseband signal prior to sampling. The sampling rate must be chosen so that there is no spectral aliasing of the sampled signal; this generally will be several times the final IF bandwidth (2–20 MHz).

Most low-cost receivers use 1-bit quantization of the digitized samples, which not only is a very-low-cost method of analog-to-digital conversion but also has the additional advantage that its performance is insensitive to changes in voltage levels. Thus, the receiver needs no automatic gain control (AGC). At first glance, it would appear that 1-bit quantization would introduce severe signal distortion. However, the noise, which is Gaussian and typically much greater than the signal at this stage, introduces a dithering effect that, when statistically averaged, results in an essentially linear signal component. One-bit quantization does introduce some loss in SNR, typically about 2 dB, but in low-cost receivers, this is an acceptable tradeoff. A major disadvantage of 1-bit quantization is that it exhibits a capture effect in the presence of strong interfering signals and is therefore quite susceptible to jamming.

Typical high-end receivers use anywhere from 1.5-bit (three-level) to 3-bit (eight-level) sample quantization. Three-bit quantization essentially eliminates the SNR degradation found in 1-bit quantization and materially improves performance in the presence of jamming signals. However, to gain the advantages of multibit quantization, the ADC input signal level must exactly match the ADC dynamic range. Thus, the receiver must have AGC to keep the ADC input level constant. Some military receivers use even more than 3-bit quantization to extend the dynamic range so that jamming signals are less likely to saturate the ADC. Additionally, there are some instrumentation-grade software-defined GNSS receivers that use a high number of sampling bits (i.e., 14) to capture and analyze anomalous GNSS signal events and interference [13].

6.2.4 Baseband Signal Processing

Baseband signal processing refers to a collection of high-speed algorithms implemented in dedicated hardware and controlled by software that acquire and track the GNSS signal, provide measurements of code phase and carrier phase for pseudorange and carrier phase measurements, and extract the navigation data. These baseband signal processing functions are performed by the receiver channels and associated channel processing and control functions depicted in Fig. 6.1. The functionality of the code and carrier signal acquisition and tracking functions as well as the measurements that they produce will be presented.

6.3 SIGNAL ACQUISITION AND TRACKING

When a GNSS receiver is turned on, a sequence of operations must ensue before information in a GNSS signal can be accessed and used to provide a user solution. These operations are typically as follows:

1. hypothesize about the user location
2. hypothesize about which GNSS satellites are visible to the antenna
3. estimating the approximate Doppler frequency of each visible satellite
4. searching for the signal both in frequency and code phase
5. detecting the presence of a signal code and carrier and confirming detection
6. tracking the code phase
7. tracking the carrier phase
8. performing data bit synchronization
9. decoding the navigation data.

In many GNSS receiver applications, it is desirable to minimize the time from when the receiver is first turned on until the first user solution is obtained. This time interval is commonly called *time to first fix* (TTFF). Depending on receiver characteristics, the TTFF might range from several seconds to several minutes, depending upon the amount and quality of the information the GNSS receivers has when it begins its searching process. Often the initial conditions of a GNSS receiver are referred to as a "cold start," "warm start," or "hot start." While these terms are subjective and vary, generally, a cold start means that the GNSS receiver has no information pertaining to its location (or previous location), no time, and no almanac data to aid in initial acquisition. A warm start refers to when the receiver may have some of the data items previously mentioned; for example, it may have a reasonably accurate estimate of location (where it was previously turned off and was not moved substantially) and recent GNSS satellite almanac (i.e., from the previous day). A hot start is when

the receiver has an accurate initial estimate of its location, an accurate estimate of GNSS system time, and accurate GNSS almanac (or even good GNSS ephemeris); this can occur if the receiver had a momentary outage, received assistance data via A-GPS, or other network providers.

6.3.1 Hypothesize about the User Location

Most GNSS receivers, when turned off, will store their location, so that upon power up, they can use the last location, along with other data to determine parameters to help it acquire the code and carrier phase of the various GNSS satellites. This location is typically stored in memory, often supported with a battery. Receivers that have not stored their last location or have lost their last location in memory will often begin to search for satellites at some fixed location, for example, the receiver manufacture location.

In GNSS receivers that have been moved a substantial distance (e.g., to a different continent) and begin the search based on their last stored location, searching can be substantially long as the receiver is searching for satellites that may be on the other side of the earth. In these cases, it is often best to load the user location into the receiver or to clear the almanac and allow the receiver to begin a new acquisition process.

6.3.2 Hypothesize about Which GNSS Satellites Are Visible

An important consideration in minimizing the TTFF is to avoid a fruitless search for those satellite signals that may be blocked by the earth, are unhealthy, or are unavailable. A receiver can restrict its search to only those satellites that are visible if it knows its approximate location (even within several hundred miles) and approximate time (within approximately 10 min) and has satellite almanac data obtained within the last several months. The approximate location can be manually entered by the user or it can be the position obtained by the receiver when it was last in operation. The approximate time can also be entered manually, but most receivers have a sufficiently accurate real-time clock that operates continuously, even when the receiver is off.

Using the approximate time, approximate position, and almanac data, the receiver calculates the elevation angle of each satellite and identifies the visible satellites as those whose elevation angle is greater than a specified value, called the *mask angle*, which has typical values of 5°–20°. At elevation angles below the mask angle, atmospheric and multipath delays tend to make the signals unreliable.

Most receivers automatically update the almanac data when in use, but if the receiver is just "out of the box," or has not been used for many months, it will need to search "blind" (i.e., cold start) for a satellite signal to collect the needed almanac. In this case, the receiver will not know which satellites are visible, so it simply must work its way down a predetermined list of satellites

until a signal is found. Although such a "blind" search may take an appreciable length of time, it is infrequently needed.

6.3.3 Signal Doppler Estimation

The TTFF can be further reduced if the approximate Doppler shifts of the visible satellite signals are known. This permits the receiver to establish a frequency search pattern in which the most likely frequencies of reception are searched first. The expected Doppler shifts can be calculated from knowledge of approximate user position, approximate time, and position estimates of the satellites using valid almanac data. The greatest benefit is obtained if the receiver has a reasonably accurate clock reference oscillator.

However, once the first satellite signal is found, a fairly good estimate of receiver clock frequency error can be determined by comparing the predicted Doppler shift with the measured Doppler shift. This error can then be subtracted out while searching in frequency for the remaining satellites, thus significantly reducing the range of frequencies that need to be searched.

6.3.4 Search for Signal in Frequency and Code Phase

There are various techniques available and used in GNSS receivers to acquire and track the GNSS signal carrier phase and code phase. Traditional techniques involve sequentially searching for the desired GNSS signal in frequency and code delay. While some techniques are more popular than others, some amount of searching and confirmation is required. The traditional searching techniques will be emphasized here. Since GNSS signals are radio signals, one might assume that they could be received simply by setting a dial to a particular frequency, as is done with AM and FM broadcast band receivers. Unfortunately, this is not the case.

GNSS signals are *spread-spectrum* signals in which the codes spreads the total signal power over a wide bandwidth. The signals are therefore virtually undetectable unless they are *despread or correlated* with a replica code in the receiver that is precisely aligned with the received code. Since the signal cannot be detected until alignment has been achieved, a search over the possible alignment positions (code phase search) is required.

Almost all current GNSS receivers are multichannel units in which each channel is assigned a satellite pseudorandom noise (PRN) code and carrier frequency, and processing in the channels is carried out simultaneously. Thus, simultaneous searches can be made for all usable satellites when the receiver is turned on. Because the search in each channel consists of sequencing through the possible frequency and code delay steps in time, it is called a *sequential search*. In this case, the expected time required to acquire as many as eight satellites is typically 30–100 s, depending on the specific search parameters used.

A relatively narrow postdespreading bandwidth (perhaps 100–1000 Hz) is required to raise the SNR to detectable and/or usable levels. However, because

of the high carrier frequencies and large satellite velocities used by various GNSSs, for terrestrial users, the received signals can have large Doppler shifts (as much as ±5 kHz) with the SVs in a medium earth orbit (MEO), which may vary rapidly (by as much as 1 Hz/s). The observed Doppler shift also varies with location on Earth, so that the received frequency will generally be unknown a priori. Furthermore, the frequency error in typical receiver reference oscillators will typically cause several kilohertz or more of frequency uncertainty at the L-band. Thus, in addition to the code search, there is also the need for a search in frequency. At a given estimated Doppler and user clock error, a *frequency bin* on the order of 500 Hz can be used in the acquisition process.

Therefore, a GPS receiver must conduct a two-dimensional search in order to find each satellite signal, where the dimensions are code delay and carrier frequency uncertainty. A search must be conducted across the delay range of the code for each frequency bin searched. Depending upon the accuracy of the estimated satellite locations, user location, and time, the full code phase, or a limited code phase search, may be performed. A generic method for conducting the search is illustrated in Fig. 6.3, where the digital IF signal from the AGC is split and then multiplied by a locally generated version of the carrier (really the IF signal) in quadrature [i.e., where the signal from the ADC is multiplied by a sin function signal to produce an in-phase (I) component, and the other split ADC is multiplied by a cosine function signal to produce a quadrature (Q) component]. These I and Q components are then multiplied by delayed replicas of the code and then passed to a signal integrator (i.e., integrate and dump) circuit that has a relatively small bandwidth

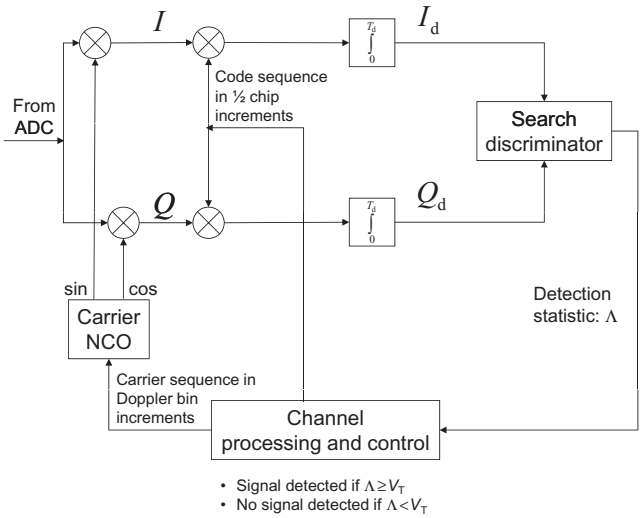


Fig. 6.3 Signal search method.

(e.g., 100–1000 Hz where the inverse represents T_d [the detection integration time shown in Fig. 6.3]). These I and Q correlation outputs are then fed into a search discriminator. While the search discriminator function may be implemented in various ways, one approach is to provide an output from the search discriminator that is proportional to $I^2 + Q^2$. The output energy of the search discriminator (Λ) serves as a signal detection statistic that can be compared to a threshold (V_T). This output will be significant only if both the selected code delay and frequency translation match that of the received signal. When the energy exceeds a predetermined threshold V_T, a tentative decision is made that a signal is being received synchronously in code phase and frequency, subject to later confirmation. The value chosen for the threshold V_T is a compromise between the conflicting goals of maximizing the probability of detecting (P_D) the signal when it is actually present at a given Doppler and code delay and minimizing the probability of false alarm (P_{FA}) when it is not.

For GNSS codes that have a relatively short period (e.g., GPS C/A-code), the unknown code delay of the signal can be considered to be uniformly distributed over its range so that each delay value is equally likely. Thus, the delays used in the code search can simply sequence from 0 to 1023.5 chips in 0.5-chip increments. For GNSS codes that have a long period, knowledge of GNSS system time, satellite location, estimated user location, and clock uncertainties (satellite and user) can help the receiver reduce the search for the correct code delay.

6.3.4.1 Sequential Searching in Code Delay For each frequency searched, the receiver generates the PRN code corresponding to the hypothesized satellite signal and moves the delay of this code in discrete steps (typically 0.5 chip) until approximate alignment with the received code (and also a match in Doppler) is indicated when the correlator output energy exceeds the threshold V_T. A step size of 0.5 code chip, which is used by many GPS receivers, is an acceptable compromise between the conflicting requirements of search speed (enhanced by a larger step size) and guaranteeing a code delay that will be located near the peak value of the code correlation function (enhanced by a smaller step size).

An important parameter in the code search is the dwell time used for each code delay position since it influences both the search speed and the detection/false-alarm performance. The dwell time is typically an integral multiple of 1 ms to assure that the correct correlation function, using the full range of code states, is obtained. Satisfactory performance is obtained with dwell times from 1 to 4 ms in most GNSS receivers that have navigation data on the respective channel, but longer dwell times are sometimes used to increase detection capability in weak-signal environments. However, if the dwell time for the search is a substantial fraction of 20 ms (the duration of one typical navigation data bit), it becomes increasingly probable that a bit transition of the 50-Hz data modulation will destroy the coherent processing of the correlator during the search and lead to a missed detection. This imposes a practical limit for a

search using coherent detection on a GNSS signal with unknown data. For signal acquisition of a dataless GNSS signal (i.e., pilot signal), no data transmission is present, so integration time is merely limited by the time spent deciding if the signal is present.

The simplest type of code search uses a fixed dwell time, a single detection threshold value V_T, and a simple yes/no binary decision as to the presence of a signal. Many receivers achieve considerable improvement in search speed by using a sequential detection technique in which the overall dwell time is conditioned on a ternary decision involving an upper and a lower detection threshold. Details on this approach can be found in the treatise by Wald [14].

6.3.4.2 Sequential Searching in Frequency The range of frequency uncertainty that must be searched is a function of the accuracy of the receiver reference oscillator, how well the approximate user position is known, the velocity of the user relative to the satellite. The first step in the search is to use stored almanac data to obtain an estimate of the Doppler shift of the satellite signal. An interval (f_{lower}, f_{upper}) of frequencies to be searched is then established. The center of the interval is located at $f_c + f_d$, where f_c is the GNSS carrier frequency to be searched and f_d is the estimated carrier Doppler shift. The width of the search interval is made large enough to account for worst-case errors in the receiver reference oscillator, in the estimate of user position, and in the user clock. Without any estimate of the Doppler shift, a typical range for the frequency search interval is $f_c \pm 5$ kHz for a terrestrial user without substantial velocity. If the user has substantial velocity, then this uncertainty should also go into the frequency uncertainty. For space-based receivers, the frequency uncertainty will likely have to be extended depending upon the user satellite orbit and relative velocity to the GNSS satellite. Extending the search window from ±5 kHz to ±20 kHz is often required.

The frequency search is conducted in N discrete frequency steps that cover the entire search interval. The value of N is $(f_{upper} - f_{lower})/\Delta f$, where Δf is the spacing between adjacent frequencies (i.e., frequency bin width). The bin width is determined by the effective bandwidth of the correlator. For the coherent processing used in many GPS receivers, the frequency bin width is approximately the reciprocal of the search dwell time. Thus, typical values of Δf are 250–1000 Hz. Assuming a ±5-kHz frequency search range, the N number of frequency steps to cover the entire search interval with a 500-Hz frequency bin would thus be 20 discrete frequency steps.

6.3.4.3 Frequency Search Strategy Because the received signal frequency is more likely to be near to, rather than far from, the Doppler estimate, the expected time to detect the signal can be minimized by starting the search at the estimated frequency and expanding in an outward direction by alternately selecting frequencies above and below the estimate. Figure 6.4 illustrates a frequency search strategy where the initial frequency offset is estimated to be 1500 Hz and a Doppler bin of 500 Hz is used over 10 steps.

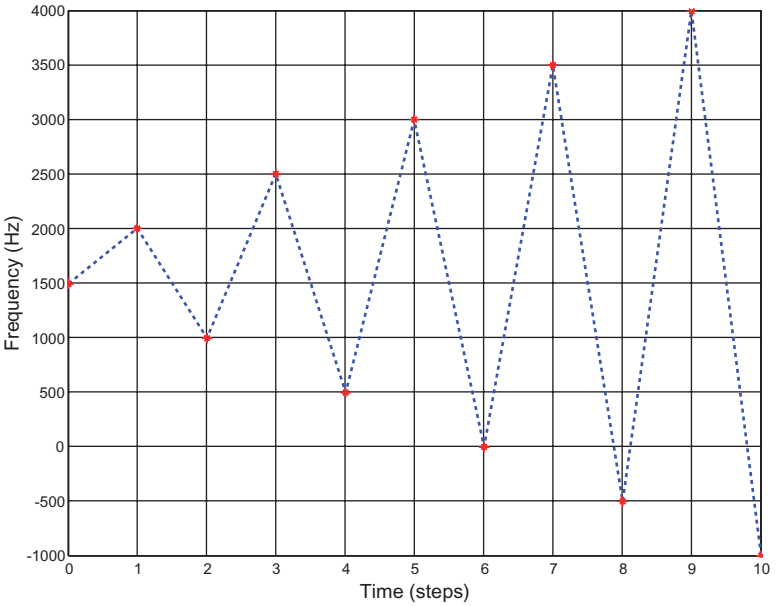


Fig. 6.4 Sequential frequency search strategy.

6.3.4.4 Parallel and Hybrid Search Methods Certain applications demand that the satellites be acquired much more rapidly (perhaps within a few seconds). This can be accomplished by using a *parallel search* technique in which extra hardware permits many frequencies and code delays to be searched at the same time. Still other techniques that take the advantages in semiconductor technology use a substantially large amount of correlation operations to process nearly all possible frequency and code delays simultaneously [5].

Other techniques involve estimating the frequency and code delay using a block of data and Fourier transform methods [15]. These types of techniques take a sample block of digital data and perform fast Fourier transform to estimate the frequency and code phase uncertainty. These uncertainties can then be used to hand over to refined time-domain tracking operations to provide a hybrid acquisition and tracking approach in a GNSS receiver [16].

6.3.5 Signal Detection and Confirmation

As previously mentioned, there is a tradeoff between the probability of detection P_D and the probability of false alarm P_{FA}. As the detection threshold V_T is decreased, P_D increases, but P_{FA} also increases, as illustrated in Fig. 6.5. Thus, the challenge in receiver design is to achieve a sufficiently large P_D so that a signal will not be missed but at the same time to keep P_{FA} small enough to avoid difficulties with false detections. When a false detection occurs, the

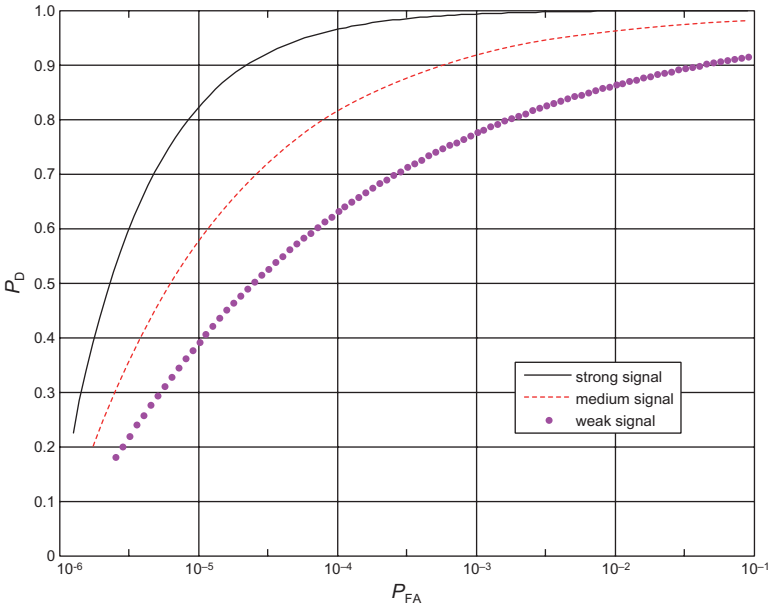


Fig. 6.5 Illustration of tradeoff between P_D and P_{FA}.

receiver will try to lock onto and track a nonexistent signal. By the time the failure to track becomes evident, the receiver will have to initiate a completely new search for the signal. On the other hand, when a detection failure occurs (i.e., a missed detection), the receiver will waste time continuing to search remaining search cells that contain no signal, after which a new search must be initiated.

6.3.5.1 Detection Confirmation One way to achieve both a large P_D and a small P_{FA} is to increase the dwell time so that the relative noise component of the detection statistic is reduced. However, to reliably acquire weak GNSS signals, the required dwell time may result in unacceptably slow search speed. An effective way around this problem is to use some form of *detection confirmation*.

To illustrate a detection confirmation process, suppose that to obtain the detection probability $P_D = 0.95$ with a typical medium-strength GPS signal, we obtain the false-alarm probability $P_{FA} = 10^{-3}$. (These are typical values for a fixed search dwell time of 3 ms.) This means that on the average, there will be one false detection in every 1000 frequency/code cells searched. Consider the following example for a typical two-dimensional GPS search region that may contain as many as 40 frequency bins and 2046 code delay positions for a GPS C/A-code for a total of $40 \times 2046 = 81{,}840$ such cells. Thus, we could expect about 82 false detections in the full search region. Given the implications of a false detection discussed previously, this is clearly unacceptable.

However, suppose that we change the rules for what happens when a detection (false or otherwise) occurs by performing a confirmation of detection before turning the signal over to the tracking loops. Because a false detection takes place only once in 1000 search cells, it is possible to use a much longer dwell (or a sequence of repeated dwell) for purposes of confirmation without markedly increasing the overall search speed, yet the confirmation process will have an extremely high probability of being correct. In the event that confirmation indicates no signal, the search can continue without interruption by the large time delay inherent in detecting the failure to track. In addition to using longer dwell times, the confirmation process can also perform a *local search* in which the frequency/code cell size is smaller than that of the main, or *global*, search, thus providing a more accurate estimate of signal frequency and code phase when detection is confirmed. Figure 6.6 depicts this process. The global search uses a detection threshold, V_T, which provides a high P_D and a moderate value of P_{FA}. Whenever the detection statistic Λ exceeds V_T at a frequency/delay cell, a confirmation search is performed in a local region surrounding that cell. The local region is subdivided into smaller cells to obtain better frequency delay resolution, and a longer dwell time is used in forming the detection statistic Λ. The longer dwell time makes it possible to use a value of V_T that provides both a high P_D and a low P_{FA}.

Some GNSS receivers use a simple adaptive search in which shorter dwell times are first used to permit rapid acquisition of moderate to strong signals

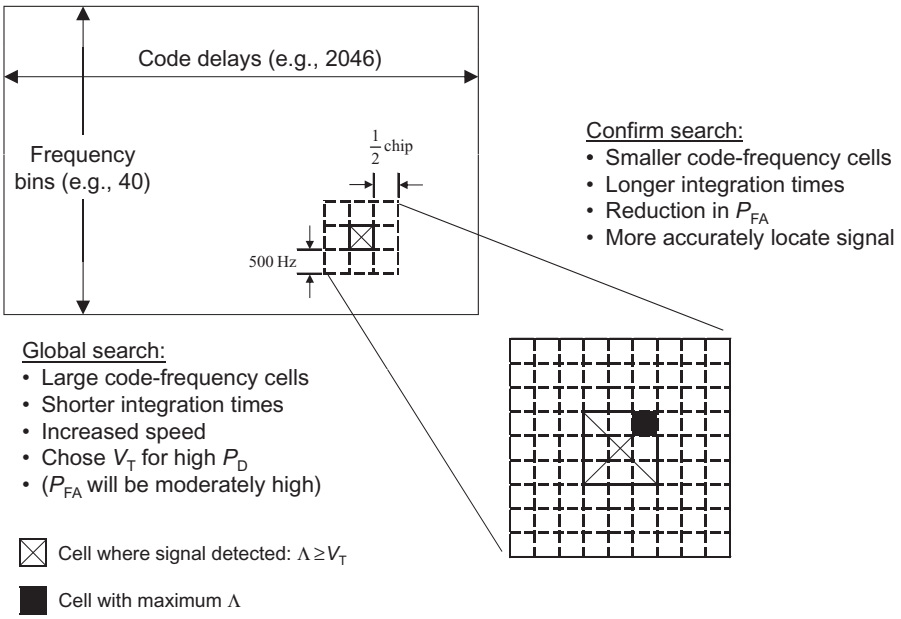


Fig. 6.6 Global and confirmation search regions.

(e.g., high elevation satellites). Whenever a search for a particular satellite is unsuccessful, or it is known that the satellite is at a low elevation angle, it is likely that the signal from that satellite is relatively weak, so the receiver can increase the dwell time and start a new search that is slower but has better performance in acquiring weak signals.

6.3.5.2 Coordination of Frequency Tuning and Code Chipping Rate As the receiver is tuned in frequency during searching, it is advantageous to advance or retard the code chipping rate of the receiver-generated code so that it is in accordance with the carrier Doppler shift under consideration. The relationship between carrier Doppler shift and the advance or retard of code chipping rate (i.e., precession rate) is given by $p(t) = f_d/N_{cyc}$, where $p(t)$ is the code precession rate in chips per second, f_d is the carrier Doppler shift (Hz), and N_{cyc} represents the number of carrier cycles in one chip of the respective code (e.g., 1540 for GPS L1 C/A-code). Here, a positive precession rate is interpreted as an increase in the chipping rate. Precession is not required while searching because the dwell times are so short. However, when detection of the signal occurs, it is important to match the incoming and reference code rates during the longer time required for detection confirmation and/or initiation of code tracking to take place.

6.3.6 Code Tracking Loop

At the time of detection confirmation, the receiver-generated local reference code will be in approximate alignment with that of the received signal (usually within 0.5 chip), and the reference code chipping rate will be approximately that of the received signal. Additionally, the frequency of the received signal will be known to within the frequency bin width Δf. However, unless further measures are taken, the residual Doppler on the received signal will eventually cause the received and reference codes to drift out of alignment and the signal frequency to drift outside the frequency bin at which detection occurred. If the code alignment error exceeds one chip in magnitude, the incoming signal will no longer despread and will disappear below the noise level. (Additionally, the correlated signal will also decrease in value as the signal frequency rate and locally generated frequency rate become less similar, and will eventually disappear if it drifts outside the detection frequency bin.) Thus, there is the need to continually adjust the timing of the locally generated reference code so that it maintains accurate alignment with the received code, a process called *code tracking*. The process of maintaining accurate frequency tuning to the received signal carrier, called *carrier tracking*, is also necessary and will be discussed in following sections.

Code tracking is initiated as soon as signal detection is confirmed, and the goal is to make the receiver's locally generated code line up with incoming code as precisely as possible. There are two objectives in maintaining alignment:

1. *Code Correlation (i.e., Signal Despreading).* The first objective is to fully despread the received signal so that it is no longer below the noise and so that information contained in the carrier and the underlying navigation data can be recovered.
2. *Pseudorange Measurements.* The second objective is to enable precise measurement of the time of arrival (TOA) of the received code for purposes of producing a pseudorange measurement. Such measurements cannot be made directly from the received signal since it is below the noise level. Therefore, a code tracking loop, which has a large processing gain, is employed to generate a reference code precisely aligned with that of the received signal. This enables pseudorange measurements to be made using the reference code instead of the much noisier received signal code waveform.

Figure 6.7 illustrates a generic code tracking loop within the receiver channels shown in Fig. 6.1. The digitized IF signal, which has a wide bandwidth due to the spreading code modulation, is completely obscured by noise. The signal power is raised above the noise power by *dispreading (or code correlation)*, where the digitized IF signal is multiplied by a receiver-generated replica of the code precisely time-aligned with the code of the received signal. The code tracking loop shown in Fig. 6.7 works together with the carrier tracking loop to iteratively acquire and track the received GNSS signal. The carrier numerically controlled oscillator (NCO) is typically multiplied by the split received

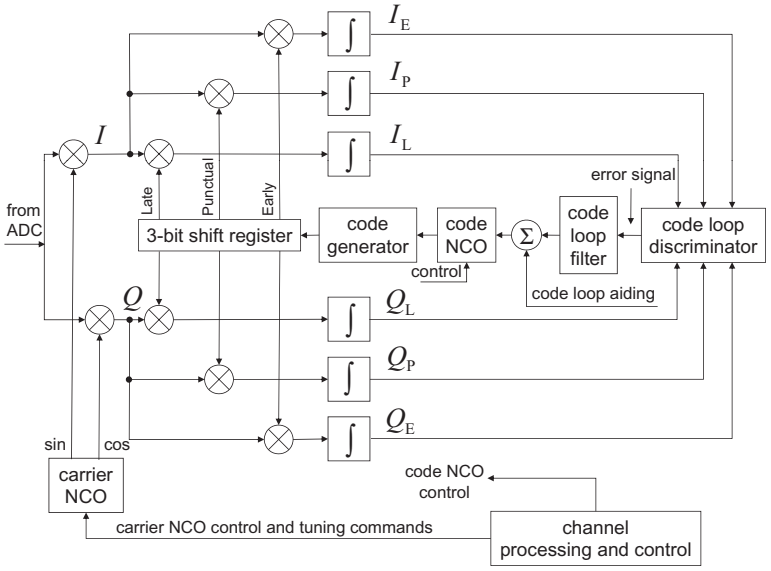


Fig. 6.7 Generic GNSS receiver code tracking loop.

digital IF signal, as is done in the carrier tracking loop shown in Fig. 6.3, to produce individual baseband I and Q signals. These I and Q signals are then typically multiplied by a delayed version of the locally generated spreading code that is controlled by the code NCO. The code tracking loop in Fig. 6.7 is known as a delay-lock loop (DLL) whereby early (E), punctual (P), and late (L) versions of the locally generated spreading code are produced to be used in the correlation process. In typical GNSS receivers, the early and late codes, respectively, lead and lag the punctual code by 0.05–1.0 code chips and maintain these relative positions during the code tracking process. These versions are used to produce delayed versions of the correlated signal for the purpose of code tracking. (Strictly speaking, the code tracking loop only tracks the code phase, and the carrier tracking loop tracks the carrier phase and performs data recovery, which will be discussed later in this chapter, but there are significant functional components that are the same.)

The mixer and integrator in each correlator channel are performed in parallel. The output magnitude of each correlator is proportional to the cross correlation of its received and reference codes, where the cross-correlation function has the triangular shaped function, with its peak occurring when the two codes are aligned. The integration process that is performed prior to the code loop discriminator is controlled by a predetection integration interval (PDI) that determines the integration time (i.e., the number of samples to integrate based on the sample rate). The amount of time the samples can be integrated will depend upon the duration of a navigation data bit, if navigation data bit synchronization has occurred, if the navigation data has been removed, or is not present for a dataless GNSS signal (i.e., pilot signal). If the navigation data has been synchronized and removed, often referred to as "data wipe-off," then the predetection integration times can exceed the data bit boundaries and enable better acquisition and tracking performance in low SNR environments. If the data transitions have not been removed, or are still unknown, data removal can be done within the code loop discriminator functional block by squaring, taking the absolute value, or extracting the envelope of each of the correlation functions (i.e., output from each of the integrators).

When attempting to acquire a GNSS signal that is encoded with navigation date and the code tracking loop is first turned on, the integration time T for the correlators is usually no more than a few milliseconds, in order to minimize corruption of the correlation process by data bit transitions of the navigation data bit stream whose locations in time are not yet known. However, after bit synchronization has located the data bit boundaries, the integration interval can span a full data bit (e.g., 20 ms for the GPS C/A-code) in order to achieve a maximum contribution to processing gain. When first attempting to acquire a GNSS signal that is not encoded with navigation, the PDI time T for the correlators can be made very long, depending upon the dynamics of the platform.

Various discriminator functions can be used in the code tracking process. Figure 6.7 depicts a generic code tracking loop that can be operated in a

noncoherent or coherent fashion. The noncoherent DLL utilizes the early and late correlator channels to form a discriminator function that produces an error signal, $e_c(\tau)$, which can be used as a basis to "speed up" or "slow down" the locally generated codes so that the locally generated punctual code is driven to alignment with the incoming signal. A common early minus late noncoherent discriminator function is shown in Eq. 6.2 to produce a code delay error signal:

$$e_c(\tau) = \frac{\left(I_E^2 + Q_E^2\right) - \left(I_L^2 + Q_L^2\right)}{\left(I_E^2 + Q_E^2\right) + \left(I_L^2 + Q_L^2\right)}. \tag{6.2}$$

The discriminator function in Eq. 6.2 has been normalized by the power in the early and late channels. Normalizing the discriminator function is useful to help minimize the error signal variations that may result in SNR variations and if the receiver is to implement dynamic discriminator functions.

Figure 6.8 illustrates a typical open-loop error signal produced by a one-chip spaced early minus late DLL verse delay (τ) between the locally generated code and the incoming received code. The error signal is slightly rounded due to filtering with its stable operating point along the linear portion of the function close to 0 delay. When the code loop eventually locks onto the received code, the stable operating point will be between the two peaks on the nearly linear slope of the discriminator function output (i.e., this error signal).

Alignment of the locally generated punctual code with the received code is maintained by using the error signal to advance or delay the reference code

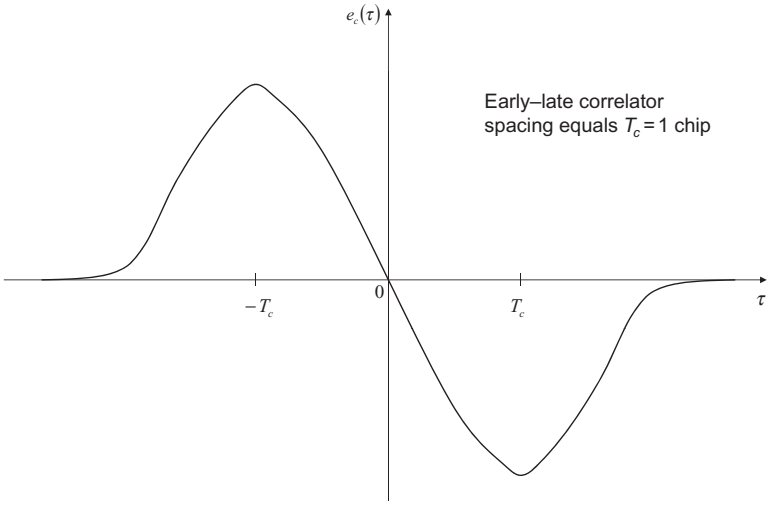


Fig. 6.8 Code tracking loop error signal (open loop).

generator rate using the discriminator error output. For example, depending upon the polarity of the slope (i.e., sensitivity) of the code NCO, when the error signal is positive, the reference code rate will speed up, and when the error signal is negative, the reference code rate will slow down. Since $e_c(\tau)$ is generally quite noisy, it is sent through a low-pass *loop filter* before it controls the clock rate of the code NCO, which drives the local reference code generator, as indicated in Fig. 6.7. The bandwidth of this filter is usually quite small, resulting in a closed-loop bandwidth typically less than 1 Hz. The bandwidth of this code loop filter can be varied based on the expected dynamics of the user platform to ensure code tracking is not lost under high dynamics, while minimizing noise in the code tracking process.

6.3.6.1 Code Loop Bandwidth Considerations The bandwidth of the code tracking loop is determined primarily by the loop filter and needs to be narrow for best ranging accuracy but wide enough to avoid loss of lock if the receiver is subject to large accelerations that can suddenly change the apparent chipping rate of the received code. Excessive accelerations cause loss of lock by moving the received and reference codes too far out of alignment before the loop can adequately respond. Once the alignment error exceeds approximately one code chip, the loop loses lock because it no longer has the ability to form the proper error signal.

In low-dynamics applications with lower-cost receivers, code tracking loop bandwidths on the order of 1 Hz permit acceptable performance in handheld units and in receivers with moderate dynamics (e.g., in automobiles). For high-dynamics applications, such as missile platforms, loop bandwidths might be on the order of 10 Hz or larger. In surveying applications, which have no appreciable dynamics, loop bandwidths can be as small as 0.01 Hz to obtain the required ranging accuracy. Both tracking accuracy and the ability to handle dynamics are greatly enhanced by means of *carrier aiding* from the receiver's carrier phase tracking loop, which will be discussed subsequently.

6.3.6.2 Coherent versus Noncoherent Code Tracking In general, the outputs of each of the correlator channels shown in Fig. 6.7 are complex (i.e., have amplitude and phase). A noncoherent code track look does not explicitly need to know the incoming carrier phase since the energies in I and Q channels are combined. Noncoherent code tracking loops will often remove the phase transitions by squaring, taking the absolute value, or rectifying the correlator outputs. Phase transitions can come about because of navigation data that are modulated within the received GNSS signal, and the data bit boundaries have not yet been determined. A distinguishing feature of a noncoherent code tracking loop is its insensitivity to the phase of the received signal. Insensitivity to phase is desirable when the loop is first turned on since, at that time, the signal phase is random and not yet under any control.

However, once the phase of the signal is being tracked, a *coherent* code tracker can be employed, in which the outputs of the early and late correlators

are purely real, and all of the signal power will be concentrated, or synchronized, with the *I* channel. In this situation, the loop error signal can be formed directly from the difference of the early and late squared magnitudes from only the *I* correlators. By avoiding the noise in the *Q* correlator outputs, a 3-dB SNR advantage is thereby gained in tracking the code.

With the code loop tracking the incoming signal, the locally generated punctual code is synchronized to the incoming coded signal. Once this occurs, not only can the *Q* channel not be used, but the energy in the P cannel can be used to help improve the performance of the code tracking loop. A discriminator function of this kind is called the dot product discriminator function [17, 18].

However, a price is paid in that the code loop error signal becomes sensitive to phase error in tracking the carrier. If phase tracking is ever lost, failure of the code tracking loop could occur (without additional aiding). This is a major disadvantage, especially in mobile applications where the signal can vary rapidly in magnitude and phase. Since noncoherent operation is much more robust in this regard and is still needed when code tracking is initiated, most GNSS receivers will use a noncoherent code tracking loop; however, some will use a hybrid approach and switch from noncoherent to coherent code tracking once synchronization has occurred.

6.3.7 Carrier Phase Tracking Loops

The purposes of tracking the carrier phase of the received GNSS signal are to

1. obtain a phase reference for coherent detection of the GNSS phase modulated data,
2. provide precise velocity measurements (via phase rate),
3. obtain integrated Doppler for rate aiding of the code tracking loop, and
4. obtain precise carrier phase measurements for use in high-accuracy receivers.

Tracking of carrier phase is usually accomplished by a phase-lock loop (PLL). A Costas-type PLL or its equivalent can be used to prevent loss of phase coherence with the received GNSS signal that has phase modulated navigation data on the GNSS carrier signal. The origin of the Costas PLL is described in Ref. 19. Figure 6.9 is a block diagram of a generic carrier tracking loop that is contained within each of the receiver channels shown in Fig. 6.1.

In Fig. 6.9, the output of the receiver IF is converted to a complex baseband signal by multiplying the signal by both the in-phase and quadrature-phase outputs of a carrier NCO. The carrier NCO produces the in-phase (i.e., sin function) and quadrature-phase (i.e., cosine function). The code is removed (assume the code tracking loop is synchronized) with the punctual code from the code tracking loop, and the resulting signal is integrated. For a GNSS

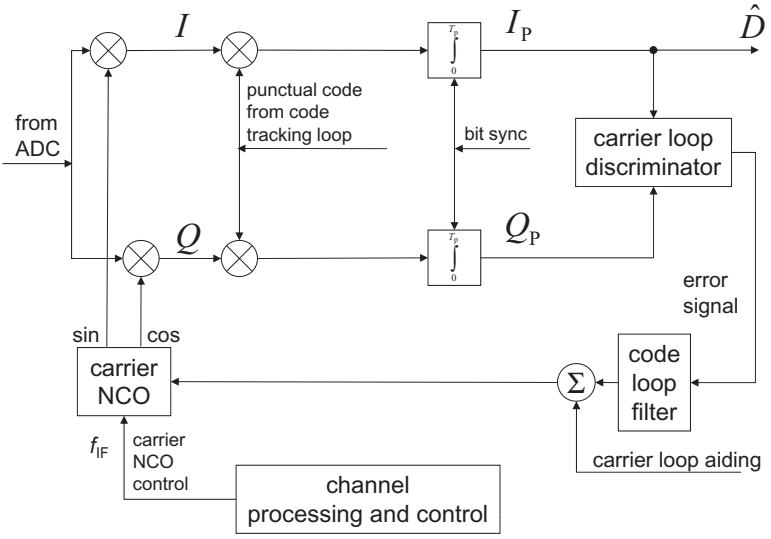


Fig. 6.9 Generic GNSS receiver carrier tracking loop.

signal that has navigation data encoded, the signal is then integrated over each navigation data bit interval to form a sequence of phasors. The phase angle of each phasor is the phase difference between the received signal carrier and the carrier NCO output during the navigation data bit integration interval. For a classical Costas loop, the discriminator function illustrated in Fig. 6.9 is a multiplier to produce a carrier loop phase error signal by multiplying together the I_p and Q_p components of each phasor. This error signal is unaffected by the phase modulated navigation data because the modulation appears on both I_p and Q_p and is removed in forming the $I_p \times Q_p$ product. After passing through a carrier low-pass loop filter, the error signal controls the carrier NCO phase to drive the loop error signal $I_p \times Q_p$ to zero (the phase-locked condition).

Because the Costas loop is unaffected by the data modulation, it will achieve phase lock at two stable points where the NCO output phase differs from that of the signal carrier by either 0° or 180°, respectively. This can be seen by considering $I = A\cos\theta$ and $Q = A\sin\theta$, where A is the phasor amplitude and θ is its phase. Then, the product of the I_p and Q_p channels is shown in Eq. 6.3:

$$\phi_e = I_p \times Q_p = A^2 \cos\theta \sin\theta = \frac{1}{2} A^2 \sin 2\theta. \tag{6.3}$$

There are ambiguous values of θ in $(0, 2\pi)$, where the error signal $I_p \times Q_p = 0$. Two of these are the stable points, namely, $\theta = 0°$ and $\theta = 180°$, toward which the loop tends to return if disturbed. Since $\sin 2\theta$ is unchanged by 180° changes

in θ caused by the data bits, the data modulation will have no effect. At either of the two stable points, the Q_p integrator output is nominally zero and the I_p integrator output contains the demodulated data stream, but with a polarity ambiguity that can be removed by observing the navigation data frame preamble. Thus, the Costas loop has the additional feature of serving as a data demodulator.

In the Costas loop design shown, the phase of the received signal is measured by comparing the phase of the carrier NCO output with a reference signal. Normally, the reference signal frequency is a rational multiple of the same crystal-controlled oscillator that is used in frequency shifting the GNSS signal down to the last IF. When the carrier NCO is locked to the phase of the incoming signal, the measured phase rate will typically be in the range of ±5 kHz due to signal Doppler shift. Two types of phase measurements are usually performed on a periodic basis (the period might be every navigation data interval). The first is an accurate measurement of the phase modulo 2π, which is used in precision carrier phase ranging. The second is the number of cycles (including the fractional part) of phase change that have occurred from a defined point in time up to the present time. The latter measurement is often called integrated Doppler and is used for aiding the code tracking loop. By subtracting consecutive integrated Doppler measurements, extremely accurate average frequency measurements can be made, which can be used by the navigation filter to accurately determine user velocity.

Although the Costas loop is not disturbed by the presence of data modulation, at low SNR, its performance degrades considerably from that of a loop designed for a pure carrier. The degradation is due to the noise × noise component of the $I_p \times Q_p$ error signal. Furthermore, the data bit duration of the I_p and Q_p integrations represents a limit to the amount of coherent processing that can be achieved. If it is assumed that the maximum acceptable bit error rate (BER) for the navigation data demodulation is 10^{-5}, GPS signals typically become unusable when C/N_0 falls below about 25 dB-Hz.

There are various carrier phase discriminator functions that can be implemented to successfully track the carrier phase. While the classical Costas loop discriminator function merely multiplies the punctual I_P and Q_P channel together and applies the error signal to the carrier loop filter, an alternative optimum PLL discriminator function takes the punctual I_P and Q_P channels and implements a full four quadrant arctan function (i.e., atan 2), which can be implemented with a memory lookup table; for this phase discriminator, the baseband phase is defined by Eq. 6.4:

$$\phi_e = \text{atan } 2(Q_P, I_P). \tag{6.4}$$

The design bandwidth of the PLL is determined by the SNR, desired tracking accuracy, signal dynamics, and ability to "pull in" when acquiring the signal or when the lock is momentarily lost.

6.3.7.1 PLL Capture Range An important characteristic of the PLL is the ability to pull in to the frequency of a received signal. When the PLL is first turned on, assuming code acquisition, the difference between the signal carrier frequency and the carrier NCO frequency must be sufficiently small or the PLL will not lock. In typical GNSS applications, the PLL must have a relatively small bandwidth (1–10 Hz) to prevent loss of lock due to noise. However, this results in a small pull-in (or capture) range (perhaps only 3–30 Hz), which would require small (hence many) frequency bins in the signal acquisition search algorithm. Advanced PLL designs or use of a frequency lock loop (FLL) can enhance the PLL capture range performance.

6.3.7.2 PLL Order The *order* of a PLL refers to the number of integrators within the tracking loop and thereby its capability to track different types of signal dynamics. Most GPS receivers use second- or third-order PLLs. A second-order loop can track a constant rate of phase change (i.e., constant frequency) with zero average phase error and a constant rate of frequency change with a nonzero but constant phase error. A third-order loop can track a constant rate of frequency change with zero average phase error and a constant acceleration of frequency with nonzero but constant phase error. Most receivers typically use a second-order PLL with fairly low bandwidth because the user dynamics are minimal and the rate of change of the signal frequency due to satellite motion is sufficiently low (<1 Hz/s) such that phase tracking error is negligible. On the other hand, receivers designed for high dynamics (i.e., missiles) will sometimes use third-order or even higher-order PLLs to avoid loss of lock due to the large accelerations encountered.

The price paid for using higher-order PLLs is a somewhat lower robust performance in the presence of noise. If independent measurements of platform dynamics are available (such as accelerometer or INS outputs), they can be used to aid the PLL by reducing stress on the loop. This can be advantageous because it often renders the use of higher-order loops unnecessary.

6.3.7.3 Use of Frequency-Lock Loops (FLLs) for Carrier Capture Some receivers avoid the conflicting demands of the need for a small bandwidth and a large capture range in the PLL by using an FLL to aid the PLL. The capture range of an FLL is typically much larger than that of a PLL, but the FLL cannot lock onto nor track the actual phase of the received signal. Therefore, an FLL is often used to pull the carrier NCO frequency into the capture range of the PLL. A typical FLL design is similar to the PLL shown in Fig. 6.9, except a different discriminator function is used, and there is no navigation data output. The FLL generates a loop error signal that is approximately proportional to the rotation rate of the baseband signal phasor and is derived from the vector cross product of successive baseband phasors where a fixed delay is used, typically, 1–5 ms. If we let t_1 correspond to the outputs at a given time, and t_2 correspond to the corresponding outputs 1–5 ms later, then a maximum

likelihood (ML) estimator for the carrier FLL is shown in Eq. 6.5, where the error signal here is frequency error [20]:

$$f_e = \frac{\text{atan2}(\text{cross}, \text{dot})}{t_2 - t_1}, \tag{6.5}$$

where

$$\text{cross} = I_{P1} \times Q_{P2} - I_{P2} \times Q_{P1}$$
$$\text{dot} \quad = I_{P1} \times I_{P2} + Q_{P1} \times Q_{P2}.$$

6.3.8 Bit Synchronization

For GNSS signals that have navigation data encoded, synchronizing to the data bit edges is necessary in order to coherently integrate the punctual channel and to optimally recover the navigation data bits. Before bit synchronization can occur, the PLL must be locked to the GNSS signal. This can be accomplished by initially running a Costas loop with a relatively short integration time (e.g., 1 ms) where each interval of integration is over one period of the code (e.g., GPS C/A-code), starting and ending at the code epoch. Since the navigation data bit transitions can occur at code epochs using phase modulation, there can be no bit transitions while integration is taking place. When the PLL achieves lock, the output of the I_p integrator will change as a function of the navigation data bits. Once the navigation data bit boundaries have been determined, then the output of the punctual integrator can be integrated over the entire number of codes used to encode a single data bit (e.g., 20 for GPS C/A-code).

A simple method of bit synchronization is to clock a modulo counter with the epochs of the receiver-generated reference code and to record the count each time the polarity of the I_p integrator output changes. The modulo of the counter will be the number of code epochs that are used to encode the navigation data. For example, with the GPS C/A-code, 20 C/A-codes are transmitted for each navigation data bit; thus, a modulo 20 counter should be used in the clock recovery process. A histogram of the frequency of each count is constructed, and the count having the highest frequency identifies the epochs that mark the data bit boundaries. (Counts corresponding to lower frequencies will correspond to multiple data bits of the same polarity and are of limited value in the navigation data clock recovery process.)

6.3.9 Data Bit Demodulation

Once navigation data bit synchronization has been achieved, demodulation of the navigation data bits can occur. As previously described, many GNSS receivers demodulate the data by integrating the I_p component of the baseband phasor generated by a Costas loop, which tracks the carrier phase. Each

data bit is generated by integrating the I_p component over a number of code epochs used to encode at single navigation data bit (e.g., 20-ms interval for the GPS C/A-code) from one data bit boundary to the next. The Costas loop causes a polarity ambiguity of the data bits that can be resolved by observation of the subframe preamble in the navigation message data.

6.4 EXTRACTION OF INFORMATION FOR USER SOLUTION

After carrier, code, and data clock synchronization, the navigation data decoding can be performed to extract the navigation information encoded onto the GNSS broadcast. This navigation information will be used by the GNSS receiver to calculate several important parameters for the user solution including the determination of the following:

1. signal transmission time,
2. position and velocity of each satellite,
3. pseudorange measurements,
4. delta pseudorange measurements,
5. carrier Doppler measurements, and
6. integrated Doppler measurements.

6.4.1 Signal Transmission Time Information

The GNSS receiver can calculate the time of transmission of the GNSS signal (i.e., the GNSS time that the signal left the satellite) by decoding the GNSS system time encoded onto the broadcast and counting the number of code cycles and chips that have lapsed until the receiver has correlated to the spreading code. In particular, for GPS, the *Z*-count represents the number of 1.5 s X1 epochs that have elapsed since the beginning of the week. The time of week (TOW), part of the *Z*-count, is encoded at the beginning of every subframe and represents a GPS system time reference to the beginning of the next subframe. (The receiver will still need to apply the SV transmitter clock error correction to correct the actual SV clock used to this reference time encoded as the TOW count.) After the code and carrier tracking loops have synchronized to the received signal, the receiver will then count the whole number of code epochs, the whole number of code chips, and the fractional number of code chips that have lapsed since that *Z*-count, from the code NCO (with respect to the punctual code). This count is performed within the GNSS receiver by a function that is commonly referred to as the *code accumulator*. The information that goes into the code accumulator count is obtained from the navigation data and the code NCO. (The part of the code accumulator that counts the whole and fractional code chips since the last code epoch is often referred to as the *code state*, which can be obtained from the code NCO.) Thus,

the total number (whole and fractional) of code chips that have lapsed since the *Z*-count is used to calculate the time of transmission (t_T).

6.4.2 Ephemeris Data for Satellite Position and Velocity

The ephemeris data permit the position and velocity of each satellite to be computed at the signal transmission time. The calculations are outlined in Table 3.2.

6.4.3 Pseudorange Measurements Formulation Using Code Phase

In an ideal system, with no clock, atmospheric, or measurement errors, finding the three-dimensional position of a user would consist of determining the *true range*, that is, the distance of the user from each of three or more satellites having known positions in space, and mathematically solving for a point in space where that set of ranges would occur. The range to each satellite can be determined by measuring how long it takes for the signal to propagate from the satellite to the receiver and multiplying the propagation time by the speed of light.

Unfortunately, this method of computing range would require very accurate synchronization of the satellite and receiver clocks used for the time measurements. GNSS satellites use very accurate and stable atomic clocks that are corrected from the supporting ground control segment, but it is often impractical to provide a comparable clock in a receiver. The problem of user clock synchronization is circumvented in GPS by treating the receiver clock error as an additional unknown in the navigation equations and using measurements from an additional satellite to provide enough equations for a user solution for time as well as for position. Thus, the receiver can use an inexpensive clock to make its measurements. Since the user clock error is common to all of the measurements, it will eventually cancel in the user solution. Such an approach leads to perhaps the most fundamental measurement made by a GNSS receiver, the *pseudorange* measurement from SV *i*, computed as shown in Eq. 6.6:

$$\rho_i = c\left(t_R - t_{Ti}\right), \tag{6.6}$$

where

c = GNSS propagation constant (i.e., speed of light) (m/s)
t_R = time of reception (s)
t_{Ti} = time of transmission from SVi (s).

In Eq. 6.6, t_R is the time at which a specific, identifiable portion of the signal is received; this is often derived from the reference clock at a particular sample time. The t_{Ti} is calculated from the code accumulator, with reference to the GNSS system time encoded in the broadcast (i.e., *Z*-count for GPS), which is

the time that same portion of the signal was transmitted, that is currently being correlated in the code tracking loop for SV *i*. The GNSS propagation constant is essentially the speed of light (2.99792458×10^8 m/s), as defined in the GNSS Interface Specification [21]. It is important to note that t_R is measured according to the receiver clock, which may have a large time error, which is one of the reasons why the code phase measurement produced by the receiver is called a pseudorange rather than a range measurement. Additionally, the raw pseudorange measurement will also contain the transmitter clock error, but this can largely be removed by applying the SV transmitter clock error corrections encoded in the broadcast.

Figure 6.10 shows the pseudorange measurement concept with four GNSS satellites. The raw pseudorange measurements are simultaneous snapshots at time t_R of the states of the received codes from the illustrated four satellites. This is accomplished indirectly by observation of the receiver's locally generated code state from each code tracking loop. The code state is a real number (whole and fractional chips since the last code epoch interval or the GNSS reference time for a long PRN code that has not repeated since the last GNSS reference time). For example, for the GPS C/A-code, the code state will be in the interval (0, 1023), and this will be added to the C/A-code epochs (i.e., multiples of 1-ms C/A-code epochs) within a subframe to get back to the edge of the GPS subframe, where the GPS subframe reference time is encoded as the *Z*-count (1.5-s X1 epochs). For a long PRN code that does not repeat since the GNSS reference time (e.g., GPS *Z*-count), the code accumulator (and hence code state) will represent the whole and fractional chips since the GPS reference time. In Fig. 6.10, the time of epoch edge to each SV *i*, (t_{ei}) represents

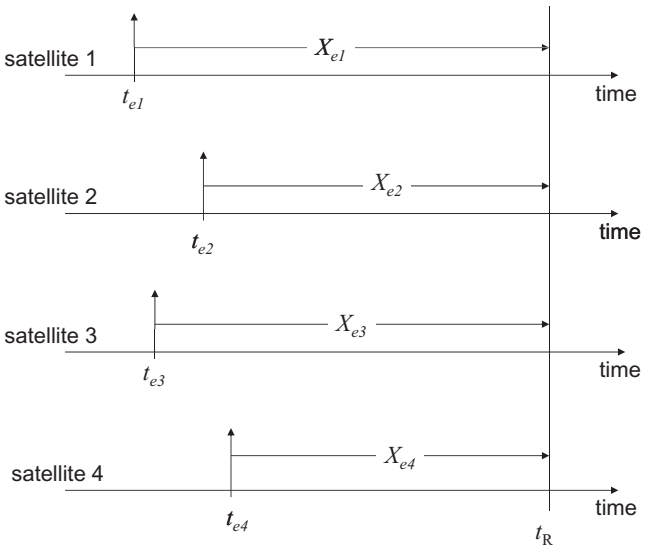


Fig. 6.10 Pseudorange measurement concept.

the time at the last code epoch edge (i.e., the beginning of the code state count [integer and fraction]). Thus, the time of transmission can be calculated as shown in Eq. 6.7:

$$t_{Ti} = t_{ei} - \frac{X_{ei}}{R_c}, \tag{6.7}$$

where

t_{ei} = time of epoch edge (s)
X_{ei} = code state count (real number)
R_c = rate of code (1/s).

6.4.3.1 Pseudorange Positioning Equations If pseudorange measurements can be made from at least four GNSS satellites, enough information exists to solve for the unknown GNSS user position (X, Y, Z) and for the receiver clock error C_b (often called the user receiver *clock bias*), expressed in units of meters. The equations are set up by equating the measured pseudorange (ρ_i) to each satellite i with the corresponding unknown user-to-satellite distance, plus the distance error due to receiver clock bias as shown in Eq. 6.8:

$$\rho_i = \sqrt{(x_i + X)^2 + (y_i + Y)^2 + (z_i + Z)^2} + C_b \quad \text{(m)}, \tag{6.8}$$

where ρ_i denotes the measured pseudorange of the ith satellite whose position in earth-centered, earth-fixed (ECEF) coordinates at t_{Ti} is (x_i, y_i, z_i) and i = 1, 2, 3 up to n, where $n \geq 4$ is the number of satellites observed. The unknowns in this nonlinear system of equations are the user position (X, Y, Z) in ECEF coordinates and the receiver clock bias C_b.

6.4.4 Measurements Using Carrier Phase

Although pseudorange measurements using the code are the most commonly employed because they provide an unambiguous range measurement, a much higher level of measurement precision can be obtained by measuring the received carrier phase of the GNSS signal. Because the carrier waveform has a very short period (6.35×10^{-10} s [or 19.0 cm] at the L1 frequency), the noise-induced error in measuring signal delay by means of phase measurements is typically 10–100 times smaller than that encountered in code delay measurements.

However, carrier phase measurements are highly ambiguous because phase measurements are simply modulo 2π numbers. Without further information, such measurements determine only the fractional part of the pseudorange when measured in carrier wavelengths. Additional processing is required to

effectively remove the carrier cycle ambiguity in the carrier phase measurements. This process is often referred to as carrier cycle *ambiguity resolution*, which determines and/or removes ambiguous integer number of wavelengths in the carrier phase measurement. The relation between the ambiguous carrier phase and the unambiguous pseudoranges can be expressed as Eq. 6.9:

$$\rho_i = (\phi_i + N_i)\lambda, \tag{6.9}$$

where

ρ_i = code phase measurement, that is, pseudorange (m)
ϕ_i = carrier phase measurement (cycles)
N_i = carrier cycle ambiguity (integer)
λ = carrier wavelength (m).

The measurements required for the determination of the N_i carrier cycle ambiguities are most often differential in nature and may be from single or multifrequency GNSS measurements. Since the code measurements are unambiguous, they significantly narrow the range of admissible integer values for the N_i. Differential techniques significantly help mitigate common systematic errors. Additionally, measurements made on two or more GNSS frequencies can help improve the speed and robustness of the carrier cycle ambiguity process. For example, for GPS, the L1 and L2 signals can be used to obtain a virtual carrier frequency equal to the difference of the two carrier frequencies (1575.42 – 1227.60 = 347.82 MHz). This combination is often referred to as the GPS *wide-lane solution*. The 86.3-cm wavelength of this virtual carrier thins out the density of carrier cycle ambiguities by a factor of about 4.5, making the ambiguity resolution process much more robust. Single-frequency, various code and carrier combinations, and triple-frequency techniques can be implemented with various performance advantages in terms of accuracy, speed, integrity, and robustness of the ambiguity solution [2, 22]. A popular technique is the least-squares ambiguity decorrelation adjustment (LAMBDA) method [23]. The LAMBDA method decorrelates the double-difference measurements to enable a more efficient search for the carrier cycle ambiguities.

In GNSS receivers, the carrier phase is usually measured by sampling the phase of the locally generated carrier signal in the carrier tracking loop. In most receivers, this locally generated signal is the carrier NCO that tracks the phase of the incoming signal at an IF. The signal phase is preserved when the incoming signal is frequency down-converted. The carrier NCO is designed to provide a digital output of its instantaneous phase (or frequency) in response to a sampling signal. Phase-based measurements are made by simultaneously sampling at time t_R the phases of the carrier NCOs tracking the various satellites signals. As with all receiver measurements, the reference for the phase measurements is the receiver's clock reference oscillator.

6.4.5 Carrier Doppler Measurement

Measurement of the received carrier frequency provides information that can be used to determine the velocity vector of the user. Although this could be done by forming differences of code-based position estimates, frequency measurement is inherently much more accurate and has faster response time in the presence of user dynamics. The equations relating the measurements of Doppler shift to the user velocity are shown in Eq. 6.10:

$$f_{di} = (\mathbf{v} \cdot \mathbf{u}_i - \mathbf{v}_i \cdot \mathbf{u}_i)\frac{1}{\lambda} + f_b \quad \text{(Hz)}, \tag{6.10}$$

where

$\mathbf{v}$ = user velocity vector (m/s)
$\mathbf{u}_i$ = unit vector pointing from user to SV i (unitless)
$\mathbf{v}_i$ = SV i velocity vector (m/s)
λ = carrier wavelength (m)
f_b = user receiver clock frequency error (Hz).

In Eq. 6.10, the unknowns are the user velocity vector ($\mathbf{v}$) and the receiver reference clock frequency error (f_b), and the known quantities are the carrier wavelength, the measured Doppler shifts f_{di}, satellite velocity vectors $\mathbf{v}_i$, and unit satellite direction vectors $\mathbf{u}_i$ for each satellite index i. The unit vectors $\mathbf{u}_i$ are determined by computing the user-to-ith satellite displacement vectors $\mathbf{r}_i$ and normalized to unit length as shown in Eq. 6.11:

$$\begin{aligned} \mathbf{r}_i &= \sqrt{(x_i + X)^2 + (y_i + Y)^2 + (z_i + Z)^2} \quad \text{(m)} \\ u_i &= \frac{\mathbf{r}_i}{|\mathbf{r}_i|} \quad \text{(unitless)}. \end{aligned} \tag{6.11}$$

In the expressions of Eq. 6.11, the ith satellite position (x_i, y_i, z_i) at time t_{Ti} is computed from the ephemeris data, and the user position (X, Y, Z) can be determined from the user position solution of the basic positioning equations using the unambiguous code.

In GNSS receivers, the Doppler measurements f_{di} are usually derived by sampling the frequency setting of the carrier NCO that tracks the phase of the incoming signal. An alternate method is to count the output cycles of the carrier NCO over a relatively short time period, perhaps 1 s or less. However, in either case, the measured Doppler shift is not the raw measurement itself but the deviation from what the nominally carrier NCO measurement would be without any signal Doppler shift, assuming that the receiver reference clock oscillator had no error.

6.4.6 Integrated Doppler Measurements

Integrated Doppler can be defined as the number of carrier cycles of Doppler shift that have occurred in a given interval ($t0, t$). For the ith satellite, the relation between integrated Doppler F_{di} and Doppler shift f_{di} is given by Eq. 6.12:

$$F_{di}(t) = \int_{t0}^{t} f_{di}\,dt. \tag{6.12}$$

However, accurate calculation of integrated Doppler according to this relation would require that the Doppler measurement be a continuous function of time. Instead, GNSS receivers use the output of the carrier NCO in the carrier tracking loop shown previously in Fig. 6.9. As shown in the carrier tracking loop of Fig. 6.9, the carrier loop error signal is filtered by the carrier loop filter (with optional aiding) and then applied to the carrier NCO. (The nominal carrier rate [scaled to IF] is commanded by the channel processing and command function.) The incremental carrier NCO increments are accumulated, and the number of whole carrier cycles (N_{cc}) that have occurred since initial time can then be counted directly, and the fractional cycles (ϕ_{cc}) are determined. Thus, the integrated Doppler measurement in a GNSS receiver is the addition of the whole carrier cycles plus the fractional cycle since initialization. (The carrier phase integration [a.k.a. accumulation] is similar to the code accumulation performed to determine the time of transmission.) Thus, the raw integrated Doppler measurement (a.k.a., carrier phase measurements) can be represented as shown in Eq. 6.13:

$$\phi_i = (N_{cc,i} + \phi_{cc,i}) \quad \text{(cycles)},$$

where

$N_{cc,i}$ = whole carrier cycle (cycles)
$\phi_{cc,i}$ = fractional carrier phase cycles (cycles),

and

$$\Phi_i = \phi_i \lambda \quad \text{(m)}, \tag{6.13}$$

where

λ = carrier wavelength (m).

Since the integrated Doppler measurements are ambiguous, they can be produced as arbitrary numbers or scaled to "look like" measurements on the order of the unambiguous pseudorange measurements. Additionally, the integrated Doppler measurements can be reported in units of (cycles) or

multiplied by the respective wavelength and reported in units of meters, as shown in the lower half of Eq. 6.13. (Note that the carrier wavelength is the nominal GNSS wavelength and does not include any Doppler for the respective SV *i*.)

Integrated Doppler measurements have several uses:

1. *Accurate Measurement of Receiver Displacement over Time.* The motion of the receiver causes a change in the Doppler shift of the incoming signal. Thus, by counting carrier cycles to obtain integrated Doppler, precise estimates of the *change* in position (*delta position*) of the user over a given time interval can be obtained. The error in these estimates is much smaller than the error in establishing the absolute position using the code measurements. Often these types of measurements are referred to as delta pseudorange measurements and are shown in Eq. 6.14, based on code phase and carrier phase, integrated Doppler measurements:

$$\Delta\rho_i(k) = \rho_i(k) - \rho_i(k - t_k) \quad \text{(m)}$$

or (6.14)

$$\Delta\Phi_i(k) = \Phi_i(k) - \Phi_i(k - t_k) \quad \text{(m)}.$$

   The capability of accurately measuring changes in position is used extensively in *real-time kinematic* application with differential GNSS. In RTK-type surveying applications, the user needs to determine the locations of many points in a given area with great accuracy (perhaps to within a few centimeters). When the receiver is first turned on, it may take a relatively long time to acquire the satellites, to make both code phase and carrier phase measurements, and to resolve carrier phase ambiguities so that the location of the first surveyed point can be determined. However, once this is done, the relative displacements of the remaining points can be found very rapidly and accurately by transporting the receiver from point to point while it continues to make integrated Doppler measurements. Most often these are done in a differential sense, and the carrier cycle ambiguities that are indeed solved for are the double-difference (or differenced) ambiguities.
2. *Positioning Based on Received Signal Phase Trajectories.* In another form of differential GPS, a fixed receiver is used to measure the integrated Doppler function, or *phase trajectory curve*, from each satellite over relatively long periods of time (perhaps 5–20 min). The position of the receiver can be determined by solving a system of equations relating the shape of the trajectories to the receiver location. The accuracy of this positioning technique, typically within a few decimeters, is not as good as that obtained by resolving the carrier cycle ambiguities but has the advantage that there is no phase ambiguity. Some handheld GPS receiv-

ers employ this technique to obtain relatively good positioning accuracy at low cost.

3. *Carrier Rate Aiding for the Code Tracking Loop.* In the code tracking loop, proper code alignment is achieved by using observations of the loop error signal to determine whether to advance or retard the state of the otherwise free-running receiver-generated code replica. Because the error signal is relatively noisy, a narrow loop bandwidth is desirable to maintain good pseudorange accuracy. However, this degrades the ability of the loop to maintain accurate tracking in applications where the receiver is subject to substantial accelerations. The difficulty can be substantially mitigated with *carrier rate aiding*, in which the primary code advance/retard commands are not explicitly derived from the code discriminator (early–late correlator) error signal but instead are primarily derived from the Doppler-induced accumulation of carrier cycles in the integrated Doppler function. For example, with the GPS C/A-code, there are 1540 carrier cycles per C/A-code chip, so the code will therefore be advanced by precisely one chip for every 1540 cycles of accumulated count of integrated Doppler. The advantage of this approach is that, even in the presence of dynamics, the integrated Doppler can track the received code *rate* very accurately. As a consequence, the error signal from the code discriminator is largely "decoupled" from the dynamics and can be used for very small and infrequent adjustments to the code generator.
4. *Postcorrelation Carrier Smoothing of the Code Measurement.* After the pseudorange and carrier phase measurements have been produced, the code measurements can be smoothed by the carrier phase measurements [24]. Smoothing time will be limited (e.g., <100 s) for single-frequency GNSS users but can be extended substantially for multifrequency users [3, 25].

6.5 THEORETICAL CONSIDERATIONS IN PSEUDORANGE, CARRIER PHASE, AND FREQUENCY ESTIMATIONS

In a GNSS receiver, the measurement error will be limited by thermal noise so it is useful to know the best performance that is theoretically possible in its presence (without additional error sources). Theoretical bounds on errors in estimating code-based and carrier-based measurements, as well as in Doppler frequency estimates, have been developed within a branch of mathematical statistics called *estimation theory*. Using estimation theory, an estimation approach called the *method of maximum likelihood* (ML) can often approach theoretically optimum performance. ML estimates of pseudorange, carrier phase, and frequency are *unbiased*, which means that the expected value of the error due to random noise is zero.

An important lower bound on the error variance of any unbiased estimator is provided by the *Cramer–Rao bound*, and any estimator that reaches this lower limit is called a *minimum-variance unbiased estimator* (MVUE). It can be shown that at the typical SNRs encountered in GNSS, ML estimates of code-based pseudorange, carrier-based measurements, and carrier frequency are all MVUEs. Thus, these estimators are optimal in the sense that no unbiased estimator has a smaller error variance [26].

6.5.1 Theoretical Error Bounds for Code Phase Measurement

As shown in Eq. 6.15, the ML estimate τ_{ML} of signal delay based on code measurements is obtained by maximizing the cross correlation of the received code $c_{rec}(t)$ with a reference code $c_{ref}(t)$ that is an identical replica (including bandlimiting) of the received code, where (0, T) is the signal observation interval:

$$\tau_{ML} = \max_{\tau} \int_0^T c_{rec}(t) c_{ref}(t-\tau)\, dt, \tag{6.15}$$

where

$c_{rec}(t)$ = received code
$c_{ref}(t-\tau)$ = reference code.

Here we assume coherent processing for purposes of simplicity. This estimator is an MVUE, and it can be shown that the error variance of τ_{ML} (which equals the Cramer–Rao bound) is expressed in Eq. 6.16:

$$\sigma^2_{\tau_{ML}} = \frac{N_0}{\int_0^T c'^2_{rec}(t)\, dt}. \tag{6.16}$$

This is a fundamental relation that in temporal terms states that the error variance is proportional to the power spectral density N_0 of the noise and is inversely proportional to the integrated square of the derivative of the received code waveform. It is generally more convenient to use an expression for the standard deviation, rather than the variance, of delay error, in terms of the bandwidth of the code. For the GPS C/A-code, this standard deviation can be calculated as a function of the C/N_0, bandwidth, and integration time as shown in Eq. 6.17 [27]:

$$\sigma_{\tau_{ML}} = \frac{3.444 \times 10^{-4}}{\sqrt{(C/N_0)WT}}. \tag{6.17}$$

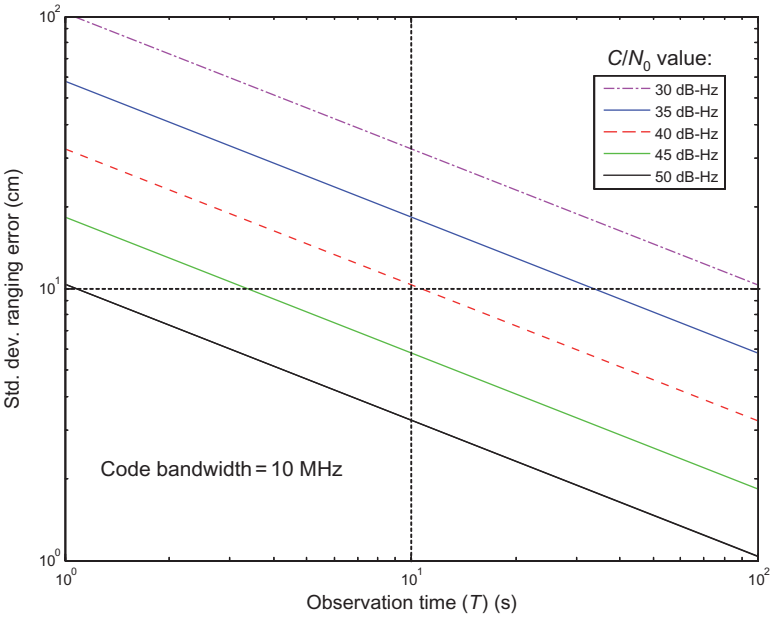


Fig. 6.11 Theoretically achievable C/A-code pseudorange error.

The expression in Eq. 6.17 assumes that the received code waveform has been bandlimited by an ideal low-pass filter with one-sided bandwidth W. The signal observation time is denoted by T, and C/N_0 is the ratio of power in the code waveform to the one-sided power spectral density of the noise. A similar expression is obtained for the error variance using the GPS P(Y)-code, except the numerator is $\sqrt{10}$ times smaller.

Figure 6.11 shows the theoretically achievable pseudoranging standard deviation error using the GPS C/A-code as a function of signal observation time for various C/N_0 values. The error is surprisingly small if the code bandwidth is sufficiently large. As an example, for a moderately strong signal with $C/N_0 = 31,623$ (45 dB-Hz), a bandwidth $W = 10$ MHz, and a signal observation time of 1 s, the standard deviation of the ML delay estimate obtained from Eq. 6.17 is about 6.2×10^{-10} s, corresponding to 18.6 cm after multiplying by the speed of light.

6.5.2 Theoretical Error Bounds for Carrier Phase Measurements

At typical GNSS SNRs, the ML estimate τ_{ML} of signal delay using the carrier phase is an MVUE, and the error standard deviation can be expressed as Eq. 6.18:

$$\sigma_{\tau_{\mathrm{ML}}} = \frac{1}{2\pi f_c \sqrt{2(C/N_0)T}}. \tag{6.18}$$

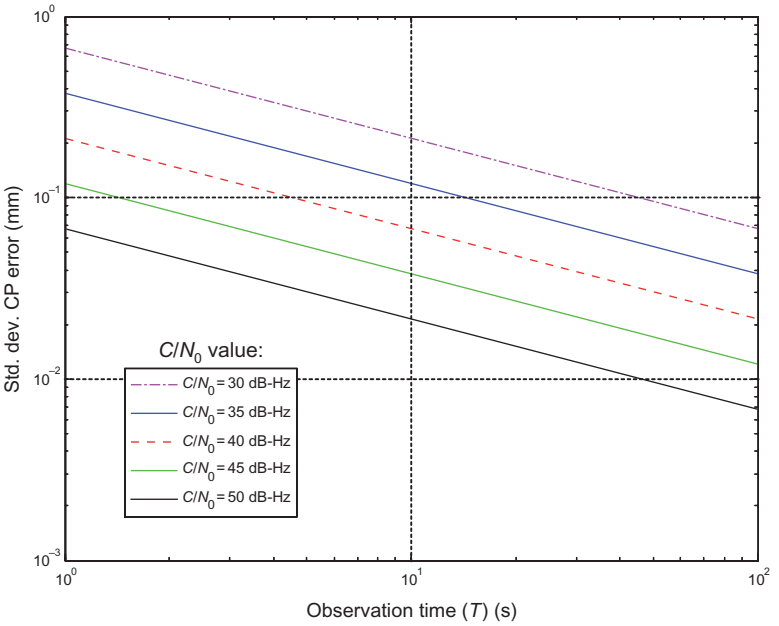


Fig. 6.12 Theoretically achievable carrier phase measurement error.

In Eq. 6.18 the GNSS carrier frequency is represented as f_c, and C/N_0 and T are the same as described for Eq. 6.17. Figure 6.12 shows the theoretically achievable carrier phase standard deviation error at the L1 frequency as a function of signal observation time for various C/N_0 values. This result is also reasonably accurate for a carrier tracking loop if T is set equal to the reciprocal of the loop bandwidth. As an example of the much greater accuracy of carrier phase measurement compared with code pseudorange measurement, a signal at $C/N_0 = 45$ dB-Hz observed for 1 s can theoretically yield an error standard deviation of 4×10^{-13} s, which corresponds to only 0.12 mm. However, typical errors of 1–3 mm are experienced in most receivers as a result of random phase jitter in the reference oscillator.

6.5.3 Theoretical Error Bounds for Frequency Measurement

The ML estimate τ_{ML} of the carrier frequency is also an MVUE, and its error standard deviation can be expressed as Eq. 6.19:

$$\sigma_{\tau_{ML}} = \frac{3}{\sqrt{2\pi^2 (C/N_0) T^3}}. \tag{6.19}$$

Figure 6.13 shows the theoretically achievable frequency estimation error as a function of signal observation time for various C/N_0 values. A 1-s observation

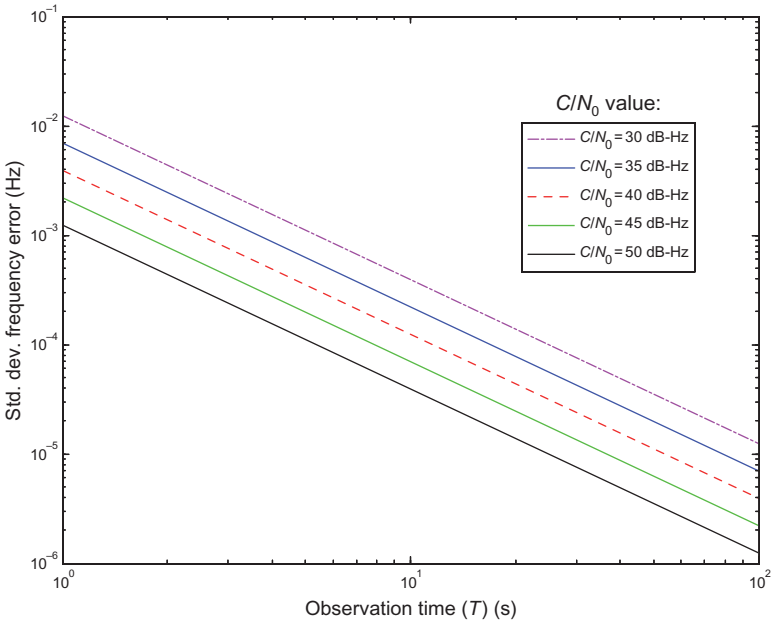


Fig. 6.13 Theoretically achievable frequency estimation error.

of a despread GNSS carrier with $C/N_0 = 45$ dB-Hz yields a theoretical error standard deviation of about 0.002 Hz, which could also be obtained with a phase tracking loop having a bandwidth of 1 Hz. As in the case of phase estimation, however, phase jitter in the receiver reference oscillator yields frequency error standard deviations from 0.05 to 0.1 Hz.

6.6 HIGH-SENSITIVITY A-GPS SYSTEMS

Over the last decade, a significant amount of emphasis and development has been made in the area of high-sensitivity GPS receivers for use in poor signal environments. More generally, such receivers can be designed and used with any GNSS, such as the GLONASS and Galileo. A major application is incorporation of such receivers in cell phones, thus enabling a user to automatically transmit its location to rescue authorities in emergency 911 (E-911) calls. The utility of using GPS for such E-911 applications has been facilitated by the removal of selective availability (SA) to GPS on May 1, 2000 and the United States mandated accuracies put into law by the U.S. Congress to support E-911 [28, 29]. Such a receiver must be able to reliably operate deep within buildings or heavy vegetation, which severely attenuates the GPS signals.

In order to achieve the reliable and rapid positioning for such applications, assisting data from a BS receiver (the server) at a location having good signal

reception are sent to the user's MS receiver (the client). The assisting data can include BS location, satellite ephemeris data, the demodulated navigation data bit stream, frequency calibration data, and timing information. In addition, the BS can provide pseudorange and/or carrier phase measurement information that enable differential operation. The assisting data can be transmitted via a cell phone or other radio links. In some cases, the assisting information can be transmitted over the Internet and relayed to a cell phone via a local-area wireless link, or the cellular network directly. Significant effort has gone into developing standards and formats to support A-GPS for cellular network and handset providers. These efforts have been coordinated by the 3rd Generation Partnership Project (3GPP) [8].

Assistance data cannot only increase the sensitivity of the client receiver but can also significantly reduce the time required to obtain a position solution. A typical stand-alone GPS receiver can acquire signals down to about –145 dBm and might require a minute or more to obtain a position from a cold start. On the other hand, high-sensitivity A-GPS receivers are currently being produced by a number of major manufacturers who are claiming sensitivities in the –155- to –165-dBm range and a cold start TTFF as low as 1 s [30]. To gain the required sensitivity and processing speed, A-GPS receivers usually capture several seconds of received signals in a memory that can be accessed at high speed to facilitate the signal processing operations.

6.6.1 How Assisting Data Improves Receiver Performance

6.6.1.1 Reduction of Frequency Uncertainty To achieve rapid positioning, the range of frequency uncertainty in acquiring the satellites at the client receiver must be reduced as much as practicable in order to reduce the search time. Reducing the number of searched frequency bins also increases receiver sensitivity because the acquisition false-alarm rate is reduced. Two ways that frequency uncertainty can be reduced are as follows:

1. *Transmission of Doppler Information.* The server *can* accurately calculate signal Doppler shifts at its location and transmit them to the user. For best results, the user must either be reasonably close to the server's receiver or must know its approximate position to avoid excessive uncompensated differential Doppler shift between server and client. For cellular applications, the BS to MS distance is usually within a couple of miles and this Doppler estimate is sufficiently accurate.
2. *Transmission of a Frequency Reference.* If only Doppler information is transmitted to the user, the frequency uncertainty of the client receiver local oscillator still remains an obstacle to rapid acquisition. Today's technology can produce oscillators that have a frequency uncertainty on the order of 1 ppm at a cost low enough to permit incorporation into a consumer product such as a cell phone. Even so, at the GNSS L1

frequency, 1 ppm translates into about ±1575 Hz of frequency uncertainty. Thus, even with an accurate Doppler estimate from the BS, the MS must search over this frequency uncertainty induced by the receiver's reference oscillator. Once the first satellite is acquired, the local oscillator offset can be determined (using the accurate Doppler estimate from the BS), and the frequency uncertainty in searching for the remaining satellites can thereby be reduced to a small value. To remedy the problem of acquiring the first satellite in a sufficiently short time, some A-GPS implementations use an accurate frequency reference transmitted from the server to the client, in addition to satellite Doppler measurements. However, this requirement complicates the design of the server-to-client communication system and is undesirable when trying to use an existing communication system for assisting purposes. If the communication system is a cell phone network, every cell tower would need to transmit a precise frequency reference or encode this timing into a message.

6.6.1.2 Determination of Accurate Time In order to obtain accurate pseudoranges, a conventional GPS receiver obtains time information from the navigation data message that permits the precise GPS time of transmission of any part of the received signal to be determined at the receiver. When a group of pseudorange measurements is made, the time of transmission from each satellite is used for two purposes: (1) to obtain an accurate position of each satellite at the time of transmission and (2) to compute the pseudorange by computing the difference between signal reception time (according to the receiver clock) and transmission time, as shown previously in Eq. 6.6.

In order to obtain time information from the received GPS signal, a conventional receiver must go through the steps of acquiring the satellite signal, tracking it in code phase and carrier phase with a PLL to form a coherent reference for data demodulation, achieve bit synchronization, demodulate the data, achieve frame synchronization, locate the portion of the navigation message that contains the GPS system time information (i.e., *Z*-count), and finally, continue to keep track of time (usually by counting code epochs as they are received).

However, it is desirable to avoid these numerous and time-consuming steps in a positioning system that must reliably obtain a position within several seconds of startup in a weak-signal environment. Because the legacy navigation data message encodes GPS time information only once per 6-s subframe, the receiver may have to wait a minimum of 6 s to obtain it (additionally, more time is needed to phase-lock to the signal and to achieve bit and frame synchronization). Furthermore, if the signals are below about –154 dBm, demodulation of the navigation data message has an error rate that may preclude the reading of time from the signal.

If the position of the client is known with sufficient accuracy (even within 100 km), it is possible to resolve the difference in times of transmission. This

is possible because the times of transmission of the C/A-code epochs are known to be integer multiples of 1 ms according to SV time (which can be corrected to GPS time using slowly changing time correction data sent from the server). This integer ambiguity in differences of time transmission is resolved by using approximate ranges to the satellites, which are calculated from the approximate position of the client and insertion of approximate time into satellite ephemeris data sent by the server to the client. For this purpose, the accuracy of the approximated time needs to be sufficiently small to avoid excessive uncertainty in the satellite positions. Generally, a time accuracy of better than 10 s will suffice for this purpose.

Once the ambiguity of the differences in transmission times has been resolved, accurate positioning is possible if the positions of the satellites at transmission time are known with an accuracy comparable to the positioning accuracy desired.

However, since the satellites are moving at a tangential orbital velocity of approximately 3800 m/s, the accuracy in knowledge of signal transmission time for the purpose of locating the satellites must be significantly more accurate than that required for the ambiguity resolution previously described.

Most weak-signal A-GPS rapid positioning systems obtain the necessary time accuracy for locating the satellites by using time information transmitted from the server. It is important to recognize that such time information must be in "real time"; that is, it must have a sufficiently small uncertainty in latency as it arrives at the client receiver. For example, a latency uncertainty of 0.1 s could result in a satellite position error of 380 m along its orbital path, causing a positioning error of the same order of magnitude. Transmission of time from the server with small latency uncertainty has an impact on the design of the server-to-client communication system such as cellular networks, to provide A-GPS positioning services.

6.6.1.3 Transmission of Satellite Ephemeris Data Due to the structure of the GPS legacy navigation message, up to 36 s (i.e., 2 x the first three subframes, worst case) is required for a stand-alone GPS receiver to obtain the ephemeris data necessary to determine the position of a satellite. This delay is undesirable in emergency applications. Furthermore, in indoor operation, the signal is likely to be too weak to demodulate the ephemeris data. The problem is solved if the server transmits the data to the client via a high-speed communication link. The Internet can even be used for this purpose if the client receiver has access to a high-speed Internet connection or the network provider can provide this information directly using the A-GPS standard messaging [8].

6.6.1.4 Provision of Approximate Client Location Some servers (e.g., a cell phone network) can transmit the approximate position of the client receiver to the user. As mentioned previously, this information can be used to resolve the ambiguity in times of signal transmission from the satellites.

6.6.1.5 Transmission of the Demodulated Navigation Bit Stream The ultimate achievable receiver sensitivity is affected by the length of the signal capture interval and the presence of navigation data modulation on the GPS signal.

Fully Coherent Processing If the GPS signal were modulated only by the C/A-code and contained no navigation data modulation, maximum theoretically possible acquisition sensitivity would result from fully coherent delay and Doppler processing. In this form of processing, the baseband signal in the receiver is frequency-shifted and precession-compensated in steps (frequency bins), and for each step, the signal is cross correlated with a replica of the C/A-code spanning the entire signal observation interval. Alternatively, the 1-ms periods of the C/A-code could be synchronously summed prior to cross correlation.

However, the presence of the 50-bps navigation data modulation precludes the use of fully coherent processing over signal capture intervals exceeding 20 ms unless some means is available to reliably strip the data modulation from the signal. If the server can send the demodulated data bit stream to the client, the data modulation can be stripped from the user's received signal, thus enabling fully coherent processing. However, the timing of the bit stream must be known with reasonable accuracy (within approximately 1 ms); otherwise, a search for time alignment must be made.

Partially Coherent Processing In the absence of a demodulated data bit stream from the server, a common method of dealing with the presence of data modulation is to coherently process the signal within each data bit interval, followed by noncoherent summation of the results. Assuming that the timing of the data bit boundaries is available from the server, the usual implementation of this technique is to first coherently sum the 20 periods of the complex baseband C/A-coded signal within each data bit. For each data bit, a waveform is produced that contains one, 1-ms period of the C/A-code, with a processing gain of $10\log_{10}(20) = 13$ dB. Each waveform is then cross correlated with a replica of the C/A-code to produce a complex-valued cross-correlation function. The squared magnitudes of the cross-correlation functions are computed and summed to produce a single function spanning 1 ms, and the location of the peak value of the function is the signal delay estimate. We shall call this form of processing *partially coherent*.

When 1 s of the received signal is observed by the user, fully coherent processing provides approximately a 3- to 4-dB improvement over partially coherent processing. It is important to note that fully coherent processing has a major drawback without assisted navigation data in that many more code delay and frequency bins must be processed, which either dramatically slows down processing speed or requires a large amount of parallel processing to maintain that speed.

Data Detection and Removal by the Client Receiver An alternate method of achieving fully coherent processing is to have the client receiver detect the data bits and use them to homogenize the polarity of the signal, thus permitting coherent processing over the full signal capture interval. In order for this method to be effective, the signal must be strong enough to ensure reliable data bit detection. Furthermore, a phase reference is needed, and it should be estimated using the entire signal observation. A practical technique for estimating phase, which approaches theoretically optimum results, is the method of ML. We shall call this methodology *coherent processing with data stripping*, or simply *data stripping* for short. (Some refer to this as *data wipe-off*.) At low signal power levels, its performance approaches that of partially coherent processing, and at high signal levels, its performance approaches that of fully coherent processing. At first glance, it seems that data stripping might give a worthwhile advantage over partially coherent processing. However, it shares a common disadvantage with fully coherent processing in that a larger number of code delay and frequency bins must be processed. Additionally, when the navigation data changes, resulting from an ephemeris update, the data change must be detected and new navigation should be used.

6.6.1.6 Server-Provided Location Some servers (e.g., a cell phone network) support a user solution for the MS. In these types of networks, raw measurement data are collected at the MS and then these raw measurement data are sent to the BS. The MS still needs to synchronize the carrier and code phase so that valid (i.e., correlated) measurement data are available. At the BS, the GPS ephemeris and error mitigation data are available, and the MS user solution is calculated. The BS then transmits back to the MS its position solution for use. This type of approach takes the advantage of utilizing the full information that may be available at the BS and supporting network and reduce computational burden on the MS.

6.6.2 Factors Affecting High-Sensitivity Receivers

In a good signal environment, a certain amount of SNR implementation loss is tolerable. Typical stand-alone GPS receivers for outdoor use may have total losses as great as 3–6 dB. However, in a high-sensitivity receiver, the maintenance of every decibel of SNR is important, thus requiring attention to minimizing losses that would otherwise not be of concern. The following are some of the more important issues that arise in high-sensitivity receiver design.

6.6.2.1 Antenna and Low-Noise RF Design A good antenna and a low-noise receiver RF front end are very important elements of a high-sensitivity receiver, the details of which were discussed in Chapter 5.

6.6.2.2 Degradation Due to Signal Phase Variations With fully coherent processing over long time intervals, performance is adversely affected by signal

phase variations from sources including Doppler curvature due to satellite motion, receiver oscillator phase stability, and motion of the receiver. Doppler curvature can be partially predicted from assisting almanac or ephemeris data, but its accuracy depends on knowledge of the approximate position of the user. On the other hand, oscillator phase noise is random and unpredictable, and hence resistant to compensation (for this reason, research efforts are currently under way to produce a new generation of low-cost atomic and optical frequency sources). Especially pernicious is motion of a GPS receiver in the user's hand. Because of the short wavelengths of GPS signals, such motion can cause phase variations of more that a full cycle during the time that the receiver is searching for a signal, thus seriously impairing acquisition performance.

6.6.2.3 Signal Processing Losses There are various forms of processing loss that should be minimized in a high-sensitivity GPS receiver.

Digitization Losses due to quantization of the ADC digital output must be minimized. The 1-bit quantization often used in low-cost receivers causes almost 2 dB of SNR loss. Hence, it is desirable to use an ADC with at least 2 bits in high-sensitivity applications.

Sampling Considerations The bandwidth of the receiver should be large enough to avoid SNR loss. However, this generally requires higher sampling rates with an attendant increase in power consumption and processing loads, a factor that is detrimental to low-cost, low-power consumer applications.

Correlation Losses Rapid signal acquisition drives the need for coarser quantization of the correlator reference code phase during signal search. However, this causes correlation loss, and an acceptable tradeoff must be made. Correlation loss is further exacerbated if the receiver bandwidth is made small to reduce the required sampling rate.

Doppler Compensation Losses One source of these losses is "scalloping loss," caused by the discrete steps of the Doppler frequencies used in searching for the satellites. Scalloping loss can be as large as 2 dB in some receivers. Another source is phase quantization of the Doppler compensation, which can introduce a degradation of as much as 1 dB in the simplest designs.

6.6.2.4 Multipath Fading It is common in poor signal environments, especially indoors, for the signal to have large and/or numerous multipath components. In addition to causing pseudorange and carrier phase biases, multipath can significantly reduce receiver sensitivity when phase cancellation of the signal occurs.

6.6.2.5 Susceptibility to Interference and Strong Signals As receiver sensitivity is decreased (i.e., gets more sensitive), so does the susceptibility to

various forms of interference. Although this is seldom a problem with receivers of normal sensitivity, in a high-sensitivity receiver, steps must be taken to prevent erroneous acquisition of lower-level PRN code correlation sidelobes from both desired and undesired satellite signals.

6.6.2.6 The Problem of Time Synchronization In an A-GPS architecture designed for rapid positioning (within a few seconds) using weak signals, the user's receiver does not have time to read the unambiguous GPS time from the received signal itself. The need for the BS to transmit to the user low-latency time accurate enough to estimate the positions of the satellites is a limiting factor.

6.6.2.7 Difficulties in Reliable Sensitivity Assessment Realistic assessment of receiver sensitivity is a challenging task. At the extremely low signal levels for which a high-sensitivity receiver has been designed, laboratory signal generators often have signal leakage, which causes the signal levels to be higher than indicated by the generator. For this and other reasons, care must be taken in the test and evaluation of high-sensitivity receivers to accurately measure their true sensitivity.

6.7 SOFTWARE-DEFINED RADIO (SDR) APPROACH

At the time GPS receivers were first developed in the mid-1970s, most were full 19-in. rack mounted units that contained analog components. It was not until the late 1970s and early 1980s when the first all-digital receiver was manufactured [31]. As technology advanced, the use of application-specific integrated circuit (ASIC) and microprocessors found application in most GNSS receiver designs. Indeed, today, most GNSS receivers and/or chip sets are a combination of ASIC and/or microprocessor-based semiconductor devices. These types of GNSS receiver architectures provide a cost-effective solution, in a small form factor with low power consumption. One disadvantage of these popular configurations is the flexibility in controlling certain aspects of the signal processing of the GNSS signals for advanced users. A more flexible GNSS architecture has emerged that is called an SDR approach. While it is true that all-digital GNSS receivers in operation today operate on some form of software, defined by the design programmer, the distinguishing feature in SDRs is that the user can control and program the details of the GNSS receiver's operation. SDRs typically operate on digitized GNSS signal samples. Most often, an RF front end is utilized and digital IF samples are processed by the SDR as illustrated in Fig. 6.1. An SDR can operate utilizing a microprocessor, a field-programmable gate array, or a processor commonly found in a personnel computer. Research in user-defined SDR materialized in the mid-1990s [12] and a limited number of SDR products are commercially

available today [32]. Most SDRs today are used as research and/or development tools.

6.8 PSEUDOLITE CONSIDERATIONS

Pseudolites (i.e., pseudosatellites) have been used in GNSS for a variety of limited applications. For new GNSS development, pseudolites can provide the range source before SVs can be launched into space. Such a system was used in the initial development of the GPS and was dubbed an "inverted range" [33]. For indoor or underground applications, pseudolites can provide a ranging source where GNSS SV signals cannot be received. Pseudolites can also have some utility to augment a GNSS when enough SV signals are not available due to signal availability [34].

Some apparent advantages of pseudolites are their availability since they are typically ground based (vice space based), their increased power available (since they are closer to the user) compared to an SV, and their geometric relationship to the user (where they are typically in a location where an SV is not, and thereby help the dilution of precision in the user solution). However, each of these attributes come with some impact on the GNSS receiver and other nonparticipating GNSS users that must be dealt with.

Although the signal level received from GNSS SVs is relatively weak, one advantage is that they are all received at approximately the same power level (e.g., –130 dBm ± 6 dB or so), where signal power level variations largely come from receiving the SV at different aspect angles. Typically, the processing gain of the code-division multiple access (CDMA) spreading code used to encode the GNSS signal can accommodate this power level variation without detrimental effects.

As for the pseudolite (PL), the power received can vary much more dramatically, depending upon application geometry and antenna pattern gains. This is mainly due to the spatial loss factor in propagation that varies as a factor of $(\lambda/4\pi R)^2$, where λ is the carrier wavelength and R is the range between the transmitter and the receiver. Usually, in pseudolite applications, the dynamic range (up to 50 dB or so from 80 m to 20 nmi [35]) needed to support the application exceeds the processing gain of the CDMA spreading code used. When the dynamic range of the application exceeds the processing gain of the CDMA code, interference can occur [36]. The interference that one code (at a strong power level) causes to another code (at a weak power level) in the correlation process is called the *near–far problem*. To mitigate these effects, pulsing the PL signal at a low duty cycle will minimize these effects on nonparticipating GNSS users [37]. Since most CDMA GNSS receivers can tolerate low duty cycle pulsed interference, they can continue to operate with minimal effect. These nonparticipating GNSS receivers will, however, see a slight reduction in their C/N_0, resulting from some GNSS code chips being lost

in the correlation process. Pseudolites based on the GPS C/A are limited by the eventual processing gain and cross-correlation performance of the code. Improvements over a C/A-code-based pseudolite can be gained by using a more random code (i.e., wideband [WB] PL [38]).

As the duty cycle of a pulsed pseudolite signal is decreased to minimize its effects on nonparticipating GNSS users, the peak power of the transmitter typically needs to be increased to maintain the same operational range for the pseudolite link (without explicit steps being taken in the GNSS/pseudolite receiver). This, along with the large dynamic range requirement, can cause saturation effects in the GNSS/pseudolite receiver. Saturation in the RF front end (mainly the mixer) and at the digital ADC level can produce measurement biases [39]. Mitigation of these effects can be done by RF AGC (depicted in Fig. 6.1) [35].

Since the pseudolite is often placed on the ground, source (i.e., ground) multipath can be a substantial error source for code-based PL applications. This can also be the case for indoor, tunnel, or underground applications of PL systems. To help mitigate these source-induced multipath errors, an advanced multipath limiting antenna (MLA) can be used [35].

While code-based pseudolite systems provide an unambiguous pseudorange, to avoid the code-based multipath and power bias, carrier phase-based pseudolite systems that rely heavily on the carrier phase measurements have been demonstrated. These types of solution techniques either solve for and remove the carrier cycle ambiguity for an absolute position [40, 41] or use a relative, that is, triple difference carrier phase user solution, initialized from an identifiable code-based user solution state (i.e., position) [38].

PROBLEMS

6.1 An ultimate limit on the usability of weak GPS signals occurs when the bit error rate (BER) in demodulating the 50-bps navigation message becomes unacceptably large. Find the signal level in dBm at the output of the receiver antenna that will give a BER of 10^{-5}. Assume an effective receiver noise temperature of 513 K, and that all signal power has been translated to the baseband I channel with optimal demodulation (integration over the 20-ms bit duration followed by polarity detection).

6.2 Support the claim that a 1-bit ADC provides an essentially linear response to a signal deeply buried in Gaussian noise by solving the following problem. Suppose that the input signal s_{in} to the ADC is a DC voltage embedded in zero-mean additive Gaussian noise $n(t)$ with standard deviation σ_{in}, and that the power spectral density of $n(t)$ is flat in the frequency interval $(-W, W)$ and zero outside the interval. Assume that the 1-bit ADC is modeled as a hard limiter that outputs a value $v_{out} = 1$ if the polarity of the signal plus noise is positive and $v_{out} = -1$ if the polarity is negative. Define the ADC output signal s_{out} by Eq. 6.20:

$$s_{\text{out}} = E[v_{\text{out}}], \tag{6.20}$$

where E denotes expectation, and let σ_{out} be the standard deviation of the ADC output. The ADC input signal-to-noise ratio SNR_{in} and output signal-to-noise ratio SNR_{out} can then be defined by Eq. 6.21:

$$SNR_{\text{in}} = \frac{s_{\text{out}}}{\sigma_{\text{in}}}$$
$$\text{and} \tag{6.21}$$
$$SNR_{\text{out}} = \frac{s_{\text{out}}}{\sigma_{\text{out}}}.$$

where s_{out} and σ_{out}, respectively, are the expected value and the standard deviation of the ADC output. Show that if $s_{\text{in}} \ll \sigma_{\text{in}}$, then $s_{\text{out}} = Ks_{\text{in}}$, where K is a constant, and

$$\frac{SNR_{\text{out}}}{SNR_{\text{in}}} = \frac{2}{\pi}. \tag{6.22}$$

Thus, the signal component of the ADC output is linearly related to the input signal component, and the output SNR is about 2 dB less than that of the input.

6.3 Some GPS receivers directly sample the signal at an IF instead of using mixers for the final frequency shift to baseband. Suppose that you wish to sample a GPS signal with a bandwidth of 1 MHz centered at an IF of 3.5805 MHz. What sampling rates will not result in frequency aliasing? Assuming that a sampling rate of 2.046 MHz was used, show how a digitally sampled baseband signal could be obtained from the samples.

6.4 Instead of forming a baseband signal with I and Q components, a single-component baseband signal can be created simply by multiplying the incoming L1 (or L2) carrier by a sinusoid of the same nominal frequency, followed by low-pass filtering. Discuss the problems inherent in this approach. (*Hint:* Form the product of a sinusoidal carrier with a sinusoidal local oscillator signal; use trigonometric identities to reveal the sum and difference frequency components, and consider what happens to the difference frequency as the phase of the incoming signal assumes various values.)

6.5 Write a computer program using MATLAB®, C or another high-level language that produces the GPS 1023-chip C/A-code used by satellite SV1. The code for this satellite is generated by two 10-stage shift registers called the *GI and G2 registers*, each of which is initialized with all 1 s. The input to the first stage of the G 1 register is the exclusive OR of its 3rd and 10th

stages. The input to the first stage of the G2 register is the exclusive OR of its 2nd, 3rd, 6th, 8th, 9th, and 10th stages. The C/A-code is the exclusive OR of stage 10 of Cil, stage 2 of 62, and stage 6 of G2. You may use the GPS IS-200F to help you in this generation.

6.6 For high accuracy of the carrier phase measurements, the most suitable carrier tracking loop will be

(a) PLL with low loop bandwidth

(b) FLL with low loop bandwidth

(c) PLL with high loop bandwidth

(d) FLL with high loop bandwidth.

6.7 Which of the following actions does not reduce the receiver noise (code)?

(a) reducing the loop bandwidth

(b) decreasing the PDI

(c) spacing the early–late correlators closer

(d) increasing the signal strength.

REFERENCES

[1] G. McGraw, "Generalized Divergence-Free Carrier Smoothing with Applications to Dual Frequency DGPS," *Navigation, Journal of the Institute of Navigation*, **56**(2), 115–122 (2009).

[2] Y. Yang, R. T. Sharpe, and R. R. Hatch, "A Fast Ambiguity Resolution Technique for RTK Embedded Within a GPS Receiver," *ION GPS 2002*, Portland, OR, September 24–27, 2002, pp. 945–952.

[3] R. R. Hatch, "A New Three-Frequency, Geometry-Free, Technique for Ambiguity Resolution," *Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2006)*, Fort Worth, TX, September 2006, pp. 309–316.

[4] R. D. J. Van Nee and J. Siereveld, "The Multipath Estimating Delay Lock Loop—Approaching Theoretical Accuracy Limits," *The Institute of Navigation GPS 1993*, Salt Lake City, UT, September 22–24, 1993.

[5] F. Van Diggelen, "Global Locate Indoor GPS Chipset & Services," *Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001)*, Salt Lake City, UT, September 2001, pp. 1515–1521.

[6] J. Ashjaee and R. Lorenz, Precision GPS Surveying After Y-Code, Ashtech, Inc., AN/AFTY/(11/92), November 1992, Magellan Corporation.

[7] RTCA, Radio Technical Commission for Maritime Services Special Committee 104. Recommended Standards for Differential GNSS (Global Navigation Satellite Systems) Service, 2011, available at: https://ssl29.pair.com/dmarkle/puborder.php?show=3, visited July 3, 2012.

[8] 3GPP, Requirements for Support of Assisted Global Positioning System (A-GPS); Frequency Division Duplex (FDD), 2012, available at: http://www.3gpp.org/ftp/Specs/html-info/25171.htm, visited July 4, 2012.

[9] L. W. Couch, III, *Digital and Analog Communications*, 7th ed. Prentice Hall, Upper Saddle River, NJ, 2007.

[10] T. Pratt, C. Bostian, and J. Allnutt, *Satellite Communications*, 2nd ed. John Wiley & Sons, Hoboken NJ, 2003.

[11] R. G. Vaughan, Member, IEEE, N. L. Scott, and D. R. White, "The Theory of Bandpass Sampling," *IEEE Transactions on Signal Processing* **39**(9), pp. 1973–1984 (1991).

[12] D. Akos, *A Software Radio Approach to Global Navigation Satellite System Receiver Design*, PhD, Ohio University, August 1997, available at: http://etd.ohiolink.edu/view.cgi?acc_num=ohiou1174615606, visited November 20, 2012.

[13] S. Gunawardena, *Development of a Transform-Domain Instrumentation Global Positioning System Receiver for Signal Quality and Anomalous Event Monitoring*, Doctoral Dissertation, Fritz J. Dolores H. Russ College of Engineering and Technology, Ohio University, June 2007.

[14] A. Wald, *Sequential Analysis*. Wiley, New York, 1947.

[15] M. Uijt de Haag, *An Investigation into the Application of Block Processing Techniques for the Global Positioning System*, PhD Dissertation, Ohio University, August 1999.

[16] F. van Graas, A. Soloviev, M. Uijt de Haag, S. Gunawardena, and M. Braasch, "Comparison of Two Approaches for GNSS Receiver Algorithms: Batch Processing and Sequential Processing Considerations," *Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2005)*, Long Beach, CA, September 2005, pp. 200–211.

[17] P. C. Fenton, W. H. Falkenberg, T. J. Ford, K. K. Ng, and A. J. Van Dierendonck, "NovAtel's GPS Receiver—The High Performance OEM Sensor of the Future," *Proceedings of the 4th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1991)*, Albuquerque, NM, September 1990, pp. 49–58.

[18] A. J. Van Dierendonck, P. Fenton, and T. Ford, "Theory and Performance of Narrow Correlator Spacing in a GPS Receiver," *Navigation, Journal of the Institute of Navigation*, **39**(3), 265–284 (1992).

[19] J. P. Costas, "Synchronous Communications," *Proceedings of the IRE 45*, 1956, pp. 1713–1718.

[20] P. W. Ward, "Performance Comparisons between FLL, PLL and a Novel FLL-Assisted-PLL Carrier Tracking Loop under RF Interference Conditions," *Proceedings of the 11th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1998)*, Nashville, TN, September 1998, pp. 783–795.

[21] GPS Directorate, Systems Engineering & Integration Interface Specification, IS-GPS-200, IS-GPS-200F, 21-Sept-2011, available at: http://www.navcen.uscg.gov/pdf/IS-GPS-200F.pdf, visited July 14, 2012.

[22] R. Hatch and T. Sharpe, "A Computationally Efficient Ambiguity Resolution Technique," *ION GPS*, 2001.

[23] P. J. G. Teunissen, P. J. de Jonge, and C. C. J. M. Tiberius, The LAMBDA-Method for Fast GPS Surveying, Presented at the International Symposium, "GPS Technology Applications," Bucharest, Romania, September 26–29, 1995.

[24] R. Hatch, "The Synergism of GPS Code and Carrier Phase Measurements," *Proceedings 3th International Geodetic Symposium on Satellite Positioning*, Las Cruces, NM, February 8–12, 1982, Vol. 2, pp. 1213–1231.

[25] P. Y. Hwang, G. A. McGraw, and J. R. Bader, "Enhanced Differential GPS Carrier-Smoothed Code Processing Using Dual-Frequency Measurements," *Navigation, Journal of the Institute of Navigation*, **46**(2), 127–138 (1999).

[26] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*, Part 1, SBN 471 89955 0, John Wiley & Sons, New York, 1968.

[27] L. Weill, "C/A Code Pseudoranging Accuracy—How Good Can It Get?" *Proceedings of the 7th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1994)*, Salt Lake City, UT, September 1994, pp. 133–141.

[28] The White House, Office of the Press Secretary, Statement by the President Regarding the United States' Decision to Stop Degrading Global Positioning System Accuracy, for immediate release May 1, 2000.

[29] FCC, FCC Amended Report to Congress on the Deployment of E-911 Phase II Services By Tier III Service Providers, Submitted Pursuant to Public Law No. 108-494, Federal Communications Commission, April 1, 2005.

[30] u-blox, Technology, Assisted GPS, 2012, available at: http://www.u-blox.com/en/assisted-gps.html, visited July 11, 2012.

[31] Institute of Navigation, Virtual Navigation Museum, Systems, SATNAV, 2012, available at: http://www.ion.org/museum/cat_view.cfm?cid=7&scid=9, visited July 11, 2012.

[32] IfEN, SX-NSR Software Receiver, 2012, available at: http://www.ifen.com/sx-nsr, visited July 11, 2012.

[33] R. L. Harrington and J. T. Dolloff, "The Inverted Range: GPS User Test Facility," *Proceedings of The Institute of Electrical and Electronics Engineers (IEEE), Position, Location, and Navigation, Symposium, (PLANS) 1976*, IEEE New York, November 1976, pp. 204–211.

[34] A. K. Brown, "A GPS Precision Approach and Landing System," *Proceedings of the 5th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1992)*, Albuquerque, NM, September 1992, pp. 373–381.

[35] C. Bartone, *Ranging Airport Pseudolite for Local Area Augmentation using the Global Positioning System*, PhD Dissertation, Ohio University, June 1998, available at: http://etd.ohiolink.edu/view.cgi?acc_num=ohiou1175095346, visited July 11, 2012.

[36] G. A. McGraw, "Analysis of Pseudolite Code Interference Effects for Aircraft Precision Approaches," *Proceedings of the 50th Annual Meeting of The Institute of Navigation*, Colorado Springs, CO, June 1994, pp. 433–437.

[37] B. Winer, et al., "GPS Receiver Laboratory RFI Tests," *Proceedings of the Institute of Navigation-GPS-96*, January 1996, pp. 674–675.

[38] S. Kiran, *A Wideband Airport Pseudolite Architecture for the Local Area Augmentation System*, PhD Dissertation, Ohio University, November 2003, available

at: http://etd.ohiolink.edu/view.cgi?acc_num=ohiou1081191846, visited July 11, 2012.

[39] S. Kiran and C. Bartone, "Verification and Mitigation of the Power-Induced Measurement Errors for Airport Pseudolites in LAAS," *GPS Solutions* **7**(4), 241–252 (2004).

[40] S. Cobb, GPS Pseudolite, Theory, Design, and Applications, Stanford University, 1997, available at: http://waas.stanford.edu/~wwu/papers/gps/PDF/Thesis/StewartCobbThesis97.pdf, visited July 11, 2012.

[41] B. Pervan, Navigation Integrity for Aircraft Precision Landing Using the Global Positioning System, Stanford University, 1996, available at: http://waas.stanford.edu/~wwu/papers/gps/PDF/Thesis/BorisPervanThesis96.pdf, visited July 11, 2012.

7

GNSS DATA ERRORS

7.1 DATA ERRORS

Ranging errors are typically grouped into six classes:

1. *Ionosphere.* Errors in corrections of pseudorange measurements caused by ionospheric effects (free electrons in the ionosphere).
2. *Troposphere.* Errors in corrections of pseudorange measurements caused by tropospheric effects; temperature, pressure, and humidity contribute to variations in the speed of light.
3. *Multipath.* Errors caused by reflected signals entering the receiver antenna.
4. *Ephemeris.* Ephemeris data errors in transmitted parameters in navigation messages for satellites' true positions.
5. *Satellite Clock.* Clock errors in the transmitted clock data for GNSS.
6. *Receiver Errors.* Errors in the receiver's measurement of range caused by thermal noise, software accuracy, and interchannel biases.

These are described in detail in Sections 7.2–7.8.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

7.2 IONOSPHERIC PROPAGATION ERRORS

The ionosphere, which extends from approximately 50–1000 km above the surface of the earth, consists of gases that have been ionized by solar radiation. The ionization produces clouds of free electrons that act as a dispersive medium for GNSS signals in which propagation velocity is a function of frequency. A particular location within the ionosphere is alternately illuminated by the sun and shadowed from the sun by the earth in a daily cycle; consequently, the characteristics of the ionosphere exhibit a diurnal variation in which the ionization is usually maximum late in midafternoon and minimum a few hours after midnight. Additional variations result from changes in solar activity.

The primary effect of the ionosphere on GNSS signals is to change the signal propagation speed as compared to that of free space. A curious fact is that the signal modulation (the code and data stream) is delayed, while the carrier phase is advanced by the same amount. Thus, the measured pseudorange using the code is larger than the correct value, while that using the carrier phase is equally smaller. The magnitude of either error is directly proportional to the total electron content (TEC) in a tube of 1 m^2 cross section along the propagation path. The TEC varies spatially due to spatial nonhomogeneity of the ionosphere. Temporal variations are caused not only by ionospheric dynamics but also by rapid changes in the propagation path due to satellite motion. The path delay for a satellite at zenith typically varies from about 1 m at night to 5–15 m during late afternoon. At low elevation angles, the propagation path through the ionosphere is much longer, so the typical corresponding delays can increase to several meters at night and as much as 50 m during the day.

Since ionospheric error is usually greater at low elevation angles, the impact of these errors could be reduced by not using measurements from satellites below a certain elevation mask angle. However, in difficult signal environments, including blockage of some satellites by obstacles, the user may be forced to use low-elevation satellites. Mask angles of 5°–10° offer a good compromise between the loss of measurements and the likelihood of large ionospheric errors.

The L1-only receivers in nondifferential operation can reduce ionospheric pseudorange error by using a model of the ionosphere broadcast by the satellites, which reduces the uncompensated ionospheric delay by about 70% on the average. During the day, errors as large as 10 m at midlatitudes can still exist after compensation with this model and can be much worse with increased solar activity. Other recently developed models offer somewhat better performance. However, they still do not handle adequately the daily variability of the TEC, which can depart from the modeled value by 25% or more.

The L1L2 receivers in nondifferential operation can take advantage of the dependence of delay on frequency to remove most of the ionospheric error. A relatively simple analysis shows that the group delay varies inversely as the square of the carrier frequency. This can be seen from the following model of the code pseudorange measurements at the L1 and L2 frequencies:

$$\rho_i = \rho \pm \frac{k}{f_i^2}, \tag{7.1}$$

where ρ is the error-free pseudorange, ρ_i is the measured pseudorange, and k is a constant that depends on the TEC along the propagation path. The subscript $i = 1, 2$ identifies the measurement at the L1 or L2 frequencies, respectively, and the plus or minus sign is identified with respective code or carrier phase measurements, respectively. The two equations can be solved for both ρ and k. The solution for ρ for ionosphere free code pseudorange measurements is

$$\rho = \frac{f_1^2}{f_1^2 - f_2^2}\rho_1 - \frac{f_2^2}{f_1^2 - f_2^2}\rho_2, \tag{7.2}$$

where f_1 and f_2 are the L1 and L2 carrier frequencies, respectively, and $\rho_1$and $\rho_2$are the corresponding pseudorange measurements.

An equation similar to Eq. 7.2 can be obtained for carrier phase measurements. However, in a nondifferential operation, the residual carrier phase error can be greater than either an L1 or L2 carrier wavelength, making ambiguity resolution difficult.

With a differential operation, ionospheric errors can be nearly eliminated in many applications because ionospheric errors tend to be highly correlated when the base and roving stations are in sufficiently close proximity. With two L1-only receivers separated by 25 km, the unmodeled differential ionospheric error is typically at the 10- to 20-cm level. At a 100-km separation, this can increase to as much as a meter. Additional error reduction using an ionospheric model can further reduce these errors by 25–50%.

7.2.1 Ionospheric Delay Model

J. A. Klobuchar's model [1, 2] for vertical ionospheric delay in seconds is given by

$$T_g = \mathrm{DC} + A\left[1 - \frac{x^2}{2} + \frac{x^4}{24}\right] \text{ for } |x| \le \frac{\pi}{2}, (\text{sec}) \tag{7.3}$$

where

$$x = \frac{2\pi(t - T_\rho)}{P}, (\text{rad})$$

DC = 5 ns (constant offset)
T_p = phase
= 50,400 s
A = amplitude
P = period

t = local time of the earth subpoint of the signal intersection with mean ionospheric height(s)

The algorithm assumes this latter height to be 350 km. The DC and phasing T_ρ are held constant at 5 ns and 14 h (50,400 s) local time.

Amplitude (A) and period (P) are modeled as third-order polynomials:

$$A = \sum_{n=0}^{3} \alpha_n \phi_m^n(s) \text{ for } A \geq 0, \text{otherwise} = 0$$

$$P = \sum_{n=0}^{3} \beta_n \phi_m^n(s) \text{ for } P > 72{,}000 \text{ s, otherwise } P = 72{,}000 \text{ s}$$

where ϕ_m is the geomagnetic latitude of the ionospheric subpoint and α_n, β_n are coefficients selected (from 370 such sets of constants) by the GPS master control station and placed in the satellite navigation upload message for downlink to the user.

A typical value of coefficients is

$$\alpha_n = \left[0.8382 \times 10^{-8}, -0.745 \times 10^{-8}, -0.596 \times 10^{-7}, -0.596 \times 10^{-7}\right],$$
$$\beta_n = \left[0.8806 \times 10^{+5}, -0.3277 \times 10^{+5}, -0.1966 \times 10^{+6}, -0.1966 \times 10^{+6}\right].$$

The parameter ϕ_m is calculated as follows with example values:

1. Subtended earth angle (EA) between user and satellite is given by the approximation

$$\text{EA} \approx \left(\frac{445}{\text{el} + 20}\right)^{-4} \text{(deg)},$$

where el is the elevation of the satellite and, with respect to the user, equals 15.5°.

2. Geodetic latitude (lat) and longitude (long) of the ionospheric subpoint are found using the approximations

$$\text{Iono lat } \phi_\text{I} = \phi_\text{user} + \text{EA} \cos \text{AZ (deg)},$$

$$\text{Iono long } \lambda_I = \lambda_\text{user} + \frac{\text{EA} \sin \text{AZ}}{\cos \phi_\text{I}} \text{(deg)},$$

$$\text{t} = 4.32 \times 10^{-4} \ \lambda_I + \text{GPS time (sec)}$$

where ϕ_user is geodetic latitude = 41°, λ_user is geodetic longitude = –73°, and AZ is azimuth of the satellite with respect to the user = 112.5°.

3. The geodetic latitude is converted to a geomagnetic coordinate system using the approximation

$$\phi_m \approx \phi_I + 11.6°(\lambda_\text{I} - 291°)(\text{deg}).$$

4. The final step in the algorithm is to account for elevation angle effect by scaling with an obliquity scale factor (SF):

$$\mathrm{SF} = 1 + 2\left[\frac{96^\circ - \mathrm{el}}{90^\circ}\right]^3 \text{ (unitless).}$$

With scaling, time delay due to ionospheric becomes

$$T_g = \begin{cases} \mathrm{SF}\left[(\mathrm{DC}) + A\left(1 - \dfrac{x^2}{2} + \dfrac{x^4}{24}\right)\right] & |x| < \dfrac{\pi}{2}, \\ \mathrm{SF}(\mathrm{DC}), & |x| < \dfrac{\pi}{2}, \end{cases}$$

$$T_G = CT_g$$

$$C = \text{speed of light}$$

$$t = \frac{\lambda_I}{15} + \mathrm{UTC},$$

where T_g is in seconds and T_G is in meters.

The MATLAB® programs Klobuchar `fix.m` and Klobuchar pseudorandom noise (PRN) for computing ionospheric delay (for PRN = satellite number) are described in Appendix A.

7.2.2 GNSS SBAS Ionospheric Algorithms

The ionospheric correction computation (ICC) algorithms enable the computation of the ionospheric delays applicable to a signal on L1 and to the GPS and wide-area reference station (WRS) L1 and L2 interfrequency biases. These algorithms also calculate grid ionospheric vertical errors (GIVEs), empirically derived error bounds for the broadcast ionospheric corrections. The ionospheric delays are employed by the space-based augmentation system (SBAS) user to correct the L1 measurements, as well as internally to correct the WRSs' L1 geostationary earth orbit (GEO) measurement for orbit determination if dual-frequency corrections are not available from GEOs. The interfrequency biases are needed internally to convert the dual-frequency-derived SBAS corrections to single-frequency corrections for the SBAS users. The vertical ionospheric delay and GIVE information is broadcast to the SBAS user via message types 18 and 26. See the Minimum Operational Performance Standards (MOPs) for details on the content and usage of the SBAS messages [3].

The algorithms used to compute ionospheric delays and interfrequency biases are based on those originated at the Jet Propulsion Laboratory [4]. The ICC models assume that ionospheric electron density is concentrated on a thin shell of height 350 km above the mean earth surface. The estimates of interfrequency biases and ionospheric delays are derived using a pair of

Kalman filters, herein referred to as the L1L2 and ionosphere (IONO) filters. The purpose of the L1L2 filter is to estimate the interfrequency biases, while the purpose of the IONO filter is to estimate the ionosphere delays. The inputs to both filters are leveled WRS receiver slant delay measurements (L2 minus L1 differential delay), which are output from the data. Both filters perform their calculations in total electron count units (TECU) (1 m of L1 ranging delay = 6.16 TECU, and 1 m of L1 − L2 differential delay = 9.52 TECU). Conceptually, the measurement equation is (neglecting the noise term)

$$\tau_{\mathrm{TECU}} = 9.52 \times \tau_m \tag{7.4}$$

$$= 9.52 \times \left(t_{\mathrm{L}2,m} - t_{\mathrm{L}1,m}\right) \tag{7.5}$$

$$= 9.52 \times \left(b^r_m + b^s_m\right) + \mathrm{TEC}_{\mathrm{TECU}} \tag{7.6}$$

$$= b^r_{\mathrm{TECU}} + b^s_{\mathrm{TECU}} + \mathrm{TEC}_{\mathrm{TECU}}, \tag{7.7}$$

where τ is the differential delay, b^r and b^s are the interfrequency biases of the respective receiver and satellite, and TEC is the ionospheric delay. The subscripts m (meters) and TECU denote the corresponding units of each term. The ionospheric delay in meters for a signal on the L1 frequency is

$$\tau^{\mathrm{L}1}_m = 1.5457 \times \frac{1}{9.52}\mathrm{TEC}_{\mathrm{TECU}} \tag{7.8}$$

$$= \frac{1}{6.16}\mathrm{TEC}_{\mathrm{TECU}}. \tag{7.9}$$

Both Kalman filters contain the vertical delays at the vertices of a triangular spherical grid of height 350 km fixed in the solar-magnetic (SM) coordinate frame as states. The L1L2 filter also contains interfrequency biases as states. In contrast, the IONO filter does not estimate the interfrequency biases, but instead they are periodically forwarded to the IONO filter, along with the variances of the estimates, from the L1L2 filter. Each slant measurement is modeled as a linear combination of the vertical delays at the three vertices surrounding the corresponding measurement pierce point (the intersection of the line of sight and the spherical grid), plus the sum of the receiver and satellite biases, plus noise. The ionospheric delays computed in the IONO filter are eventually transformed to a latitude–longitude grid that is sent to the SBAS users via message type 26. Because SBAS does not have any calibrated ground receivers, the interfrequency bias estimates are all relative to a single receiver designated as a reference, whose L1L2 interfrequency bias filter covariance is initialized to a small value, and to which no process noise is applied.

The major algorithms making up the ICC discussed here are

Initialization. The L1L2 and IONO filters are initialized using either the Klobuchar model or using previously recorded data.

Estimation. The actual computation of the interfrequency biases and ionospheric delays involves both the L1L2 and IONO filters.

Thread Switch. The measurements from a WRS may come from an alternate WRS receiver. In this case, the ICC must compensate for the switch by altering the value of the respective receiver's interfrequency bias state in the L1L2 filter. In the nominal case, an estimate of the L1L2 bias difference is available.

Anomaly Processing. The L1L2 filter contains a capability to internally detect when a bias estimate is erroneous. Both thread switch and anomaly processing algorithms may also result in the change of the reference receiver [5].

7.2.2.1 L1L2 Receiver and Satellite Bias and Ionospheric Delay Estimations for GPS

System Model For GPS, the ionospheric delay estimation Kalman filter uses a random walk system model. A state of the Kalman filter at time t_k is modeled to be equal to that state at the previous time t_{k-1}, plus a random process noise representing the uncertainty in the transition from time t_{k-1} to time t_k; that is,

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \mathbf{w}_k,$$

where $\mathbf{x}_k$ is the state vector of the Kalman filter at time t_k and $\mathbf{w}_k$ is a white process noise vector with known covariance Q. The state vector $\mathbf{x}_k$ consists of three subgroups of states: the ionospheric vertical delays at triangular tile vertices, the satellite L1L2 biases, and the receiver L1L2 biases; that is,

$$\mathbf{x}_k = \begin{bmatrix} x_{1,k} \\ \vdots \\ x_{NV,k} \\ x_{NV+1,k} \\ \vdots \\ x_{NV+NS,k} \\ x_{NV+NS+1,k} \\ \vdots \\ x_{NV+NS+NR,k} \end{bmatrix},$$

where NV is the number of triangular tile vertices, NS is the number of GPS satellites, and $N\,R$ is the number of WRSs. The values of NV, NS, and NR must be adjusted to fit the desired configuration. In simulations, one can use 24 GPS satellites in the real orbits generated by GPS Infrared Positioning System (GIPSY) using ephemeris data downloaded from the GPS bulletin board. The number of WRSs is 25 and these WRSs are placed at locations planned for SBAS operations.

Observation Model The observation model or measurement equation establishes the relationship between a measurement and the Kalman filter state vector. For any GPS satellite in view, there is an ionospheric slant delay measurement corresponding to each WRS–satellite pair. Ionospheric slant delay measurement is converted to the vertical delay at its corresponding pierce point through an obliquity factor. At any time t_k, there are approximately 80–200 pierce points, and hence the same number of ionospheric vertical delay measurements that can be used to update the Kalman filter state vector.

Denote the ionospheric vertical delay measurement at t_k for the ith satellite and jth WRS as z_{ijk}. Thus,

$$z_{ijk} = i_{ijk} + \frac{b_{si}}{q_{ijk}} + \frac{b_{sj}}{q_{ijk}} + v_{ijk},$$

where i_{ijk} is the vertical ionospheric delay at the piece point corresponding to satellite i and WRS j; b_{si} and b_{sj} are the L1L2 interfrequency biases for satellite i and WRS j, respectively; q_{ijk} is the obliquity factor; and v_{ijk} is the receiver measurement noise, white with covariance R.

To establish an observation model, we need to relate i_{ijk}, b_{si}, and b_{sj} to the state vector of the ionospheric delay estimation Kalman filter. Note that b_{si} and b_{sj} are the elements of the state vector labeled $NV + i$ and $NV + NS + j$, respectively. The relationship between i_{ijk} and the state vector is established below. The value i_{ijk} is modeled as a linear combination of the vertical delay values at the three vertices of the triangular tile in which the piece point is located, as shown in Fig. 7.1.

In Fig. 7.1, assume a pierce point P is located arbitrarily in the triangular tile ABC. The ionospheric delay at pierce point P is obtained from the vertical delay values at vertices A, B, and C using a bilinear interpolation as follows. Draw a line from point A to point P and find the intersection point D between

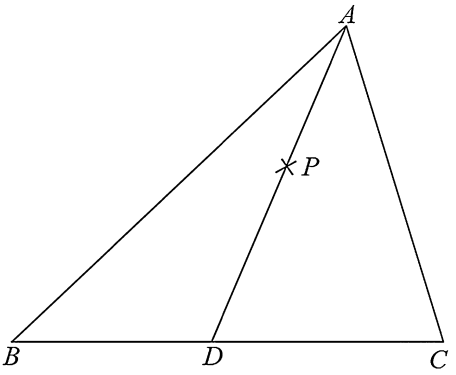


Fig. 7.1 Bilinear interpolation.

this line and the line BC. The bilinear interpolation involves two simple linear interpolations—the first yields the vertical delay value at point D from points B and C; the second yields the vertical delay value at point P from points D and A. The result can be summarized as

$$I_P = w_A I_A + w_B I_B + w_C I_C,$$

where I_P, I_A, I_B, and I_C are the ionospheric vertical delay values at points P, A, B, and C, respectively, and w_A, w_B, and w_C are the bilinear weighting coefficients from points A, B, and C, respectively, to point P. The values of w_A, w_B, and w_C can be readily calculated from the geometry involved. It is recognized that I_A, I_B, and I_C are three elements of the Kalman filter state vector. In summary, the measurement equation can be written as

$$z_{ijk} = \mathbf{h}_{ijk}\mathbf{x}_k + v_{ijk},$$

where $\mathbf{h}_{ijk}$ is the measurement matrix and v_{ijk} is the measurement noise, respectively, for the pierce point measurement for the satellite with index i and WRS with index j at time t_k. Here, $\mathbf{h}_{ijk}$ is an $(NV + NS + NR)$ dimension row vector with all elements equal to zeros except five elements. The first three of these five nonzero elements correspond to the vertices of the tile that contains the pierce point under consideration, and the other two correspond to the ith satellite and jth WRS, which yields the ionospheric slant delay measurement z_{ijk}.

UDU^T Kalman Filter (See Chapter 10) As noted previously, there are approximately 180–200 pierce points at any time t_k. Each pierce point corresponds to one of the possible combinations of a satellite and a WRS, which further corresponds to an ionospheric vertical delay measurement at that pierce point. The ionospheric estimation Kalman filter is designed so that its state vector is updated upon the reception of each ionospheric vertical delay measurement.

SM-to-Earth-Centered, Earth-Fixed (ECEF) Transformation At the end of each 5-min interval (Kalman filter cycle), the ionospheric vertical delays at the vertices of all tiles are converted from the SM coordinates to the ECEF coordinates. This conversion is completed by first transforming the SBAS ionospheric grid points (IGPs) from the ECEF coordinates to the SM coordinates. For each IGP converted to SM coordinates, the triangular tile that contains this IGP is found. A bilinear interpolation identical to the one described in Fig. 5.3 is then used to calculate the ionospheric vertical delay values at this IGP. (Transformations are given in Appendix B.)

In new GEOs (3rd, PRN 135 [Galaxy XV], at 133° W longitude; 4th, PRN 138° [Anik F1R], at 107° W longitude) will have L1L5 frequencies (see Chapter 8). Ionospheric delays can be calculated at the WRSs directly instead of using ionospheric delay provided by ionospheric grids from SBAS broadcast messages.

7.2.2.2 Kalman Filter In estimating the ionospheric vertical delays in the SM coordinate system by the Kalman filter, there are three types of estimation errors:

1. estimation error due to ionospheric slant delay measurement noise error
2. estimation error due to the temporal variation of the ionosphere
3. estimation error due to nonlinear spatial variation of the ionosphere.

Each of the three sources of error can be individually minimized by adjusting the values of the covariances **Q** and **R**. However, the requirements to minimize the errors due to noise and temporal variations are often in conflict.

Intuitively, to minimize the measurement noise implies that we want the **Q** and **R** values to result in a Kalman gain that averages out the measurement noise; that is, we want the Kalman gain to take values so that for each new measurement, the value of innovation is small, such that a relatively large noise component of the measurement results in a relatively small estimation error. On the other hand, if we want to minimize the estimation error due to temporal variations, then we want to have a Kalman gain that can produce a large innovation, so that the component in the measurement that represents the actual ionospheric delay variation with time can be quickly reflected in the new state estimate. This suggests that we usually need to compromise in selecting the values of **Q** and **R** when a conventional nonadaptive Kalman filter is used.

Although the Kalman filter estimation error is the dominant source of error, it is not the only source. The nonlinear spatial variation introduces additional error when converting the ionospheric vertical delay estimated by the Kalman filter in the SM coordinate system to the SBAS IGP in the ECEF coordinate system. This is because bilinear interpolation is used and there is an implicit assumption that interpolation is a strictly valid procedure. However, if the actual value of the vertical delay was measured at some location, it would not be equal to the value found by interpolation. Violation of this assumption results in interpolation error during the transformation. It can be shown by simulations that, under certain conditions, this conversion error can be significant and non-negligible.

In order to isolate the sources of errors and to understand how the algorithm responds to various conditions, consider seven scenarios, with each testing one aspect of the possible estimation error, and all their possible combinations.

Scenario 1: Measurement Noise In this scenario, the ionospheric vertical delay is assumed to be a time-invariant constant anywhere over the earth's surface. The Kalman filter estimation errors due to temporal and spatial variations are zero. For each of the ionospheric slant delay measurements, a zero-mean white Gaussian noise is added. The magnitude of the noise is characterized by its variance. The measurement noise is added to the slant delay rather than

the vertical delay because this is where the actual measurement noise is introduced by a GPS receiver.

Scenario 2: Temporal Variation In this scenario, the ionosphere is assumed to be uniformly distributed spatially, but its TEC values change with time; that is, the ionospheric vertical delays vary with time, but these variations are identical everywhere. Various time variation functions, such as a sinusoidal function, a linear ramp, a step function, or an impulse function, can be used to study this scenario. In a simulation using a sinusoidal time variation function, the sinusoidal function is characterized by two parameters—its amplitude and frequency. The values of these two parameters are chosen to produce a time variation that is similar in magnitude to the ionospheric delay variation data published in the literature. The measurement noise is zero. Kalman filter estimation errors due to both the measurement noise and spatial variation are fixed at zero (for this scenario).

Scenario 3: Spatial Variation In this scenario, the ionosphere is assumed to be a constant at any fixed location when observed in the SM coordinate system. The ionospheric delays at different locations in the SM coordinate system, however, are different. Various spatial variation functions can be used to study this scenario. Here, we use a three-dimensional surface constructed from two orthogonal sinusoidal functions of varying amplitudes and frequency to model the values of ionospheric vertical delays over the earth. The values of the parameters of the two sinusoidal functions are chosen to produce gradients in TEC similar in magnitude to the ionospheric delay variation data published in the literature. The measurement noise is zero. Kalman filter estimation errors due to both the measurement noise and temporal variations are fixed at zero for this scenario.

Scenario 4: Noise + Temporal Scenarios 1 and 2 are combined, and the Kalman filter estimation error due to spatial variation is zero.

Scenario 5: Noise + Spatial Scenarios 1 and 3 are combined. In this scenario, the Kalman filter estimation error due to temporal variation is zero.

Scenario 6: Temporal + Spatial Here, the Kalman filter estimation error due to measurement noise is zero. The combined values of temporal and spatial variations define the "truth ionosphere" in the simulation.

Scenario 7: Noise + Temporal + Spatial In this scenario, the parameters that define the "true ionosphere" and "measurement noise" can be configured to mimic any ionospheric conditions.

In the simulations, the GPS satellite orbits used are the precise orbits generated by GIPSY using GPS satellite ephemeris data downloaded from the GPS bulletin board. The WRS locations used are those currently recommended by

the Federal Aviation Administration (FAA). These locations may he adjusted to evaluate the impact of other WRS locations or additional WRSs.

7.2.2.3 Selection of Q and R Theoretically, a Kalman filter yields optimal estimation of the states of a system, given a knowledge of the system dynamics and measurement equations, when both the system process noise and measurement noise are zero-mean Gaussian at each epoch and white in time and their variances are known. However, in practice, the system dynamics are often unknown and system modeling errors are introduced when the actual system dynamics differ from the assumptions. In addition, the system process noise and the measurement noise are often non-Gaussian and their variances are not known precisely. To ensure a stable solution, a relatively large value of **Q** is often used, sacrificing estimation accuracy. Careful selection of **Q** and **R** values impacts the performance of the Kalman filter in practical applications, including the SBAS ionospheric estimation filter.

In each phase of the validation, many parameters are tuned. The procedures and rationale involved in selecting the final values of these parameters include an effort to distinguish those parameters for which the performance is particularly sensitive. For many parameters, performance is not particularly sensitive. Table 7.1 shows typical values of the parameters used in two Kalman filters. The L1L2 filter can be eliminated. The IONO filter, including the satellite and receiver biases, may be sufficient to estimate the biases and IONO delays. This reduces the computational load and simplifies the process.

The algorithms must be validated to ensure that the estimation accuracy is good enough to ultimately support downstream precision-approach requirements. Convergence properties of the estimation algorithms must be examined, and

TABLE 7.1. Representative Kalman Filter Parameter Values

Parameter Term	Value	Units
L1L2 filter bias process noise update interval	300	s
L1L2 filter TEC process noise	0.05	$TECU/s^{1/2}$
L1L2 filter TEC process noise update interval	300	s
Iono filter process noise	0.05	$TECU/s^{1/2}$
Iono filter process noise update interval	300	s
lono meas floor	9	$TECU^2$
Iono meas scale	0	
L1L2 filter bias process noise	4.25×10^{-4}	$TECU/s^{1/2}$
L1L2 next bias distribution time interval	300	s
L1L2 cold start bias distribution time interval	300	s
L1L2 cold start time interval	86,400	s
lono a priori covariance matrix	$400 = 20^2$	$TECU^2$
L1L2 bias a priori covariance matrix	$10{,}000 = 100^2$	$TECU^2$
(ref receiver)	10^{-10}	$TECU^2$
Maximum initial TEC	1,000	TECU
Nominal initial TEC	25	TECU

the logic associated with restarting the estimation using recorded data must be analyzed. The capabilities to perform thread switches and to detect anomalies must be examined, and the special cases necessitating a change of reference receiver. In each phase of validation, the critical test is whether there is any significant degradation in accuracy as compared to nominal performance, and whether the nominal performance itself is adequate.

7.2.2.4 Calculation of Ionospheric Delay Using Pseudoranges The calculation of ionospheric propagation delay from P-code and C/A-code can be formulated in terms of the following measurement equalities:

$$\rho_{\mathrm{RL1}} = \rho + \mathrm{L1}_{\mathrm{iono}} - c\tau_{\mathrm{RX1}} - c\tau_{\mathrm{GD}}, \tag{7.10}$$

$$\rho_{\mathrm{RL2}} = \rho + \frac{\mathrm{L1}_{\mathrm{iono}}}{(f_{\mathrm{L2}}/f_{\mathrm{L1}})^2} - c\tau_{\mathrm{RX2}} - \frac{c\tau_{\mathrm{GD}}}{(f_{\mathrm{L2}}/f_{\mathrm{L1}})^2}, \tag{7.11}$$

where

$$\left.\begin{aligned}
\rho_{\mathrm{RL1}} &= \text{L1 pseudorange}\\
\rho_{\mathrm{RL2}} &= \text{L2 pseudorange}\\
\rho &= \text{geometric distance between GPS satellite transmitter and GPS receiver, including nondispersive contributions such as tropospheric refraction and user clock error}\\
f_{\mathrm{L1}} &= \text{L1 frequency}\\
&= 1572.42\ \text{MHz (GPS)}\\
f_{\mathrm{L2}} &= \text{L2 frequency}\\
&= 1227.6\ \text{MHz (GPS)}\\
\tau_{\mathrm{RX1}} &= \text{receiver noise as manifested in code (receiver and calibration biases) at L1 (ns)}\\
\tau_{\mathrm{RX2}} &= \text{receiver noise as manifested in code (receiver and calibration biases) at L2 (ns)}\\
\tau_{\mathrm{GD}} &= \text{satellite group delay (interfrequency bias)}\\
c &= \text{speed of light}\\
&= 0.299792458\,\frac{\mathrm{m}}{\mathrm{ns}}\\
\mathrm{L1}_{\mathrm{iono}} &= \text{delay at L1 (m)}
\end{aligned}\right\}. \tag{7.12}$$

Subtracting Eq. 7.10 from Eq. 7.11, we get

$$\mathrm{L1}_{\mathrm{iono}} = \frac{\rho_{\mathrm{RL1}} - \rho_{\mathrm{RL2}}}{1-(f_{\mathrm{L1}}/f_{\mathrm{L2}})^2} - \frac{c(\tau_{\mathrm{RX1}} - \tau_{\mathrm{RX2}})}{1-(f_{\mathrm{L1}}/f_{\mathrm{L2}})^2} - c\tau_{\mathrm{GD}}. \tag{7.13}$$

What is actually measured in the ionospheric delay is the sum of receiver bias and interfrequency bias. The biases are determined and taken out from the ionospheric delay calculation. These biases may be up to 10 ns (3 m) [6, 7].

However, the presence of ambiguities N_1 and N_2 in carrier phase measurements of L1 and L2 preclude the possibility of calculating the ionosphere delay directly and would involve the solution of the carrier cycle ambiguities iteratively.

The MATLAB® program Iono_delay(PRN#) (described in Appendix A) uses pseudorange and carrier phase data from L1 and L2 signals.

7.3 TROPOSPHERIC PROPAGATION ERRORS

The lower part of the earth's atmosphere is composed of dry gases and water vapor, which lengthen the propagation path due to refraction. The magnitude of the resulting signal delay depends on the refractive index of the air along the propagation path and typically varies from about 2.5 m in the zenith direction to 10–15 m at low satellite elevation angles. The troposphere is nondispersive at the GNSS frequencies, so that delay is not frequency dependent. In contrast to the ionosphere, tropospheric path delay is consequently the same for code and carrier signal components. Therefore, this delay cannot be measured by utilizing both L1 and L2 pseudorange measurements, and either models and/or differential techniques must be used to reduce the error.

The refractive index of the troposphere consists of that due to the dry-gas component and the water vapor component, which respectively contribute about 90% and 10% of the total delay. Knowledge of the temperature, pressure, and humidity along the propagation path can determine the refractivity profile, but such measurements are seldom available to the user. However, using standard atmospheric models for the dry delay permits determination of the zenith delay to within about 0.5 m and with an error at other elevation angles that approximately equals the zenith error times the cosecant of the elevation angle. These standard atmospheric models are based on the laws of ideal gases and assume spherical layers of constant refractivity with no temporal variation and an effective atmospheric height of about 40 km. Estimation of dry delay can be improved considerably if surface pressure and temperature measurements are available, bringing the residual error down to within 2–5% of the total.

The component of tropospheric delay due to water vapor (at altitudes up to about 12 km) is much more difficult to model because there is considerable spatial and temporal variation of water vapor in the atmosphere. Fortunately, the wet delay is only about 10% of the total, with values of 5–30 cm in continental midlatitudes. Despite its variability, an exponential vertical profile model can reduce it to within about 2–5 cm.

In practice, a model of the standard atmosphere at the antenna location would be used to estimate the combined zenith delay due to both wet and dry components. Such models use inputs such as the day of the year and the

latitude and altitude of the user. The delay is modeled as the zenith delay multiplied by a factor that is a function of the satellite elevation angle. At zenith, this factor is unity, and it increases with decreasing elevation angle as the length of the propagation path through the troposphere increases. Typical values of the multiplication factor are 2 at 30° elevation angle, 4 at 15°, 6 at 10°, and 10 at 5°. The accuracy of the model decreases at low elevation angles, with decimeter level errors at zenith and about 1 m at 10° elevation.

Much research has gone into the development and testing of various tropospheric models. Excellent summaries of these appear in the literature [8–10].

Although a GNSS receiver cannot measure pseudorange error due to the troposphere, differential operation can usually reduce the error to small values by taking advantage of the high spatial correlation of tropospheric errors at two points within 0–100 km on the earth's surface. However, exceptions often occur when storm fronts pass between the receivers, causing large gradients in temperature, pressure, and humidity.

7.4 THE MULTIPATH PROBLEM

Multipath propagation of the GNSS signal is a dominant source of error in positioning, especially in differential GNSS architectures. Objects in the vicinity of a receiver antenna (notably the ground) can easily reflect GNSS signals, resulting in one or more secondary propagation paths. These secondary-path signals, which are superimposed on the desired direct-path signal, always have a longer propagation time and can significantly distort the amplitude and phase of the direct-path signal.

Errors due to multipath cannot be reduced by the use of differential GNSS since they depend on local reflection geometry near each receiver antenna. In a receiver without multipath protection, C/A-code ranging errors of 10 m or more can be experienced. Multipath cannot only cause large code ranging errors but can also severely degrade the ambiguity resolution process required for carrier phase ranging such as that used in precision surveying applications.

Multipath propagation can be divided into two classes: static and dynamic. For a stationary receiver, the propagation geometry changes slowly as the satellites move across the sky, making the multipath parameters essentially constant for perhaps several minutes. However, in mobile applications, there can be rapid fluctuations in fractions of a second. Therefore, different multipath mitigation techniques are generally employed for these two types of multipath environments. Most current research has been focused on static applications, such as surveying, where greater demand for high accuracy exists. For this reason, we will concentrate our attention to the static case.

7.4.1 How Multipath Causes Ranging Errors

To facilitate an understanding of how multipath causes ranging errors, several simplifications can be made that in no way obscure the fundamentals involved.

We will assume that the receiver processes only the C/A-code and that the received signal has been converted to complex (i.e., analytic) form at baseband (nominally zero frequency), where all Doppler shift has been removed by a carrier tracking phase-lock loop. It is also assumed that the GNSS navigation data modulation has been removed from the signal, which can be achieved by standard techniques. When no multipath is present, the received waveform is represented by

$$r(t) = ae^{j\phi}c(t-\tau)+n(t), \tag{7.14}$$

where $c(t)$ is the normalized, undelayed C/A-code waveform as transmitted; r is the signal propagation delay; a is the signal amplitude; ϕ is the carrier phase; and $n(t)$ is the Gaussian receiver thermal noise having flat power spectral density. Pseudoranging consists of estimating the delay parameter τ. As we have previously seen, an optimal estimate (i.e., a minimum-variance unbiased estimate) of τ can be obtained by forming the cross-correlation function

$$R(\tau) = \int_{T_1}^{T_2} r(t)c_r(t-\tau)dt, \tag{7.15}$$

of $r(t)$ with a replica $c_r(t)$ of the transmitted C/A-code and choosing as the delay estimate that value of τ that maximizes this function. Except for an error due to receiver thermal noise, this occurs when the received and replica waveforms are in time alignment. A typical cross-correlation function without multipath for C/A-code receivers having a 2-MHz precorrelation bandwidth is shown by the solid lines Fig. 7.2 (these plots ignore the effect of noise, which would add small random variations to the curves).

If multipath is present with a single secondary path, the waveform of Eq. 7.14 changes to

$$r(t) = ae^{j\phi_1}c(t-\tau_1)+be^{j\phi_2}c(t-\tau_2)+n(t), \tag{7.16}$$

where the direct and secondary paths have respective propagation delays τ_1 and τ_2, amplitudes a and b, and carrier phases ϕ_1 and ϕ_2. In a receiver not designed expressly to handle multipath, the resulting cross-correlation function will now have two superimposed components, one from the direct path and one from the secondary path. The result is a function with a distortion depending on the relative amplitude, delay, and phase of the secondary-path signal, as illustrated at the top of Fig. 7.2 for an in-phase secondary path and at the bottom of Fig. 7.2 for an out-of-phase secondary path. Most importantly, the location of the peak of the function has been displaced from its correct position, resulting in a pseudorange error.

In vintage receivers employing standard code tracking techniques (early and late codes separated by one C/A-code chip), the magnitude of pseudorange error caused by multipath can be quite large, reaching 70–80 m for a secondary-path signal one-half as large as the direct-path signal and having a relative delay of approximately 250 m. Further details can be found in Ref. 11.

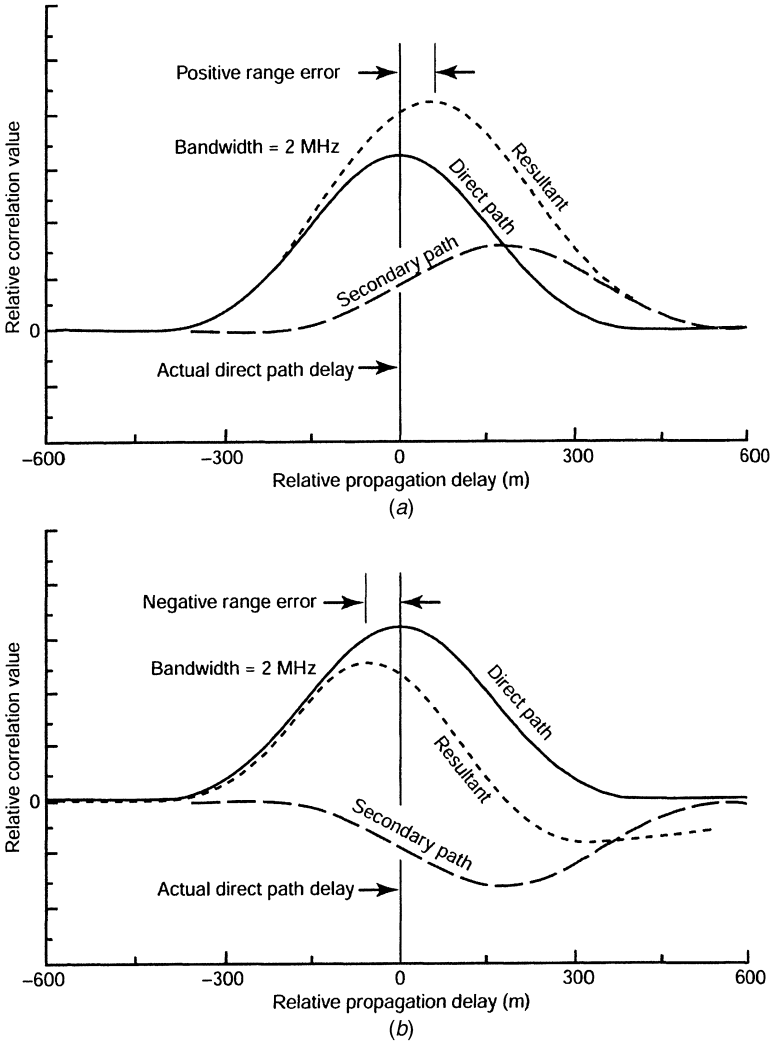


Fig. 7.2 Effect of multipath on C/A-code cross-correlation function.

7.5 METHODS OF MULTIPATH MITIGATION

Processing against slowly changing multipath can be broadly separated into two classes: spatial processing and time-domain processing. Spatial processing uses antenna design in combination with known or partially known characteristics of signal propagation geometry to isolate the direct-path received signal. In contrast, time domain processing achieves the same result by operating only on the multipath-corrupted signal within the receiver.

7.5.1 Spatial Processing Techniques

7.5.1.1 Antenna Location Strategy Perhaps the simplest form of spatial processing is to locate the antenna where it is less likely to receive reflected signals. For example, to obtain the position of a point near reflective objects, one can first use GNSS to determine the position of a nearby point "in the clear" and then calculate the relative position of the desired point by simple distance and/or angle measurement techniques. Another technique that minimizes ever-present ground signal reflections is to place the receiver antenna directly at ground level. This causes the point of ground reflection to be essentially coincident with the antenna location so that the secondary path has very nearly the same delay as the direct path. Clearly, such antenna location strategies may not always be possible but can be very effective when feasible.

7.5.1.2 Ground Plane Antennas The most common form of spatial processing is an antenna designed to attenuate signals reflected from the ground. A simple design uses a metallic ground plane disk centered at the base of the antenna to shield the antenna from below. A deficiency of this design is that when the signal wave fronts arrive at the disk edge from below, they induce surface waves on the top of the disk that then travel to the antenna. The surface waves can be eliminated by replacing the ground plane with a choke ring, which is essentially a ground plane containing a series of concentric circular troughs one-quarter wavelength deep. These troughs act as transmission lines shorted at the bottom ends so that their top ends exhibit very high impedance at the GNSS carrier frequency. Therefore, induced surface waves cannot form, and signals that arrive from below the horizontal plane are significantly attenuated. However, the size, weight, and cost of a choke ring antenna are significantly greater than those of simpler designs. Most importantly, the choke ring cannot effectively attenuate secondary-path signals arriving from above the horizontal, such as those reflecting from buildings or other structures. Nevertheless, such antennas have proven to be effective when signal ground bounce is the dominant source of multipath, particularly in GNSS surveying applications.

7.5.1.3 Directive Antenna Arrays A more advanced form of spatial processing uses antenna arrays to form a highly directive spatial response pattern with high gain in the direction of the direct-path signal and attenuation in directions from which secondary-path signals arrive. However, inasmuch as signals from different satellites have different directions of arrival and different multipath geometries, many directivity patterns must be simultaneously operative, and each must be capable of adapting to changing geometry as the satellites move across the sky. For these reasons, highly directive arrays seldom are practical or affordable for most applications.

7.5.1.4 Long-Term Signal Observation If a GNSS signal is observed for sizable fractions of an hour to several hours, one can take advantage of changes

in multipath geometry caused by satellite motion. This motion causes the relative delays between the direct and secondary paths to change, resulting in measurable variations in the received signal. For example, a periodic change in signal level caused by alternate phase reinforcement and cancellation by the reflected signals is often observable. Although a variety of algorithms have been proposed for extracting the direct-path signal component from measurements of the received signal, the need for long observation times rules out this technique for most applications. However, it can be an effective method of multipath mitigation at a fixed site, such as at a differential GNSS base station. In this case, it is even possible to observe the same satellites from one day to the next, looking for patterns of pseudorange or phase measurements that repeat daily.

Multipath Calculation from Long-Term Observations Delays can be computed as follows by using pseudoranges and carrier phases over long signal observations (one day to next). This technique may be ruled out for most applications. Ambiguities and cycle slips have been eliminated or mitigated. Let

$$\left.\begin{aligned}
\lambda_1 &= 19.03\ \text{cm, wavelength of L1 (GPS)}\\
\lambda_2 &= 24.42\ \text{cm, wavelength of L2 (GPS)}\\
\phi_{L1} &= \text{carrier phase for L1 (cycles)}\\
\phi_{L2} &= \text{carrier phase for L2 (cycles)}\\
f_{L1} &= \text{L1 frequency} = 1575.42\ \text{MHz (GPS)}\\
f_{L2} &= \text{L2 frequency} = 1227.6\ \text{MHz (GPS)}\\
\rho &= \text{error free pseudorange}\\
\rho_{RL1} &= \text{pseudorange L1 (m)}\\
\rho_{RL2} &= \text{pseudorange L2 (m)}\\
I &= \text{ionospheric delay}\\
I_{L1} &= \text{ionospheric delay in L1}\\
MP_{L1} &= \text{multipath in L1}\\
MP_{L2} &= \text{multipath in L2}
\end{aligned}\right\}; \tag{7.17}$$

for dual-frequency GNSS receivers, one obtains

$$\lambda_1\phi_{L1} = \rho - \frac{I}{(f_{L1})^2}, \tag{7.18}$$

$$\lambda_2\phi_{L2} = \rho - \frac{I}{(f_{L2})^2}. \tag{7.19}$$

Subtracting Eq. 7.19 from Eq. 7.18, one can obtain

$$\lambda_1\phi_{L1} - \lambda_2\phi_{L2} = \frac{I(f_{L1})^2 - I(f_{L2})^2}{(f_{L1})^2 (f_{L2})^2},$$

$$I_{L1} = \frac{(\lambda_1\phi_{L1} - \lambda_2\phi_{L2})(f_{L2})^2}{(f_{L1})^2 - (f_{L2})^2}, \tag{7.20}$$

$$K = \frac{(f_{L2})^2}{(f_{L1})^2 - (f_{L2})^2},$$

$$I_{L1} = K(\lambda_1\phi_{L1} - \lambda_2\phi_{L2}),$$

$$\rho_{RL1} = \rho + \frac{I}{(f_{L1})^2}. \tag{7.21}$$

Subtracting Eq. 7.18 from Eq. 7.21, one obtains the multipath as

$$MP_{L1} = \rho_{RL1} - \lambda_1\phi_{L1} - 2I_{L1}, \tag{7.22}$$

where

$$I_{L1} = \frac{I}{(f_{L1})^2}.$$

Substitute Eq. 7.20 into Eq. 7.22 to obtain

$$\begin{aligned} MP_{L1} &= \rho_{RL1} - \lambda_1\phi_{L1} - 2K(\lambda_1\phi_{L1} - \lambda_2\phi_{L2}) \\ &= \rho_{RL1} - [(1-2K)\lambda_1\phi_{L1} + 2K\lambda_2\phi_{L2}]. \end{aligned} \tag{7.23}$$

7.5.2 Time-Domain Processing

Although time-domain processing against GNSS multipath errors has been the subject of active research for at least two decades, there is still much to be learned, both at theoretical and practical levels. Most of the practical approaches have been developed by receiver manufacturers, who are often reluctant to explicitly reveal their methods. Nevertheless, enough information about multipath processing exists to gain insight into its recent evolution.

7.5.2.1 Narrow-Correlator Technology (1990–1993) The first significant means to reduce GPS multipath effects by receiver processing made its debut in the early 1990s. Until that time, most receivers had been designed with a 2-MHz precorrelation bandwidth that encompassed most, but not all, of the GPS C/A spread-spectrum signal power. These receivers also used one-chip spacing between the early and late reference C/A-codes in the code tracking loops. However, the 1992 paper [12] makes it clear that using a significantly larger bandwidth combined with much closer spacing of the early and late reference codes would dramatically improve the ranging accuracy both with

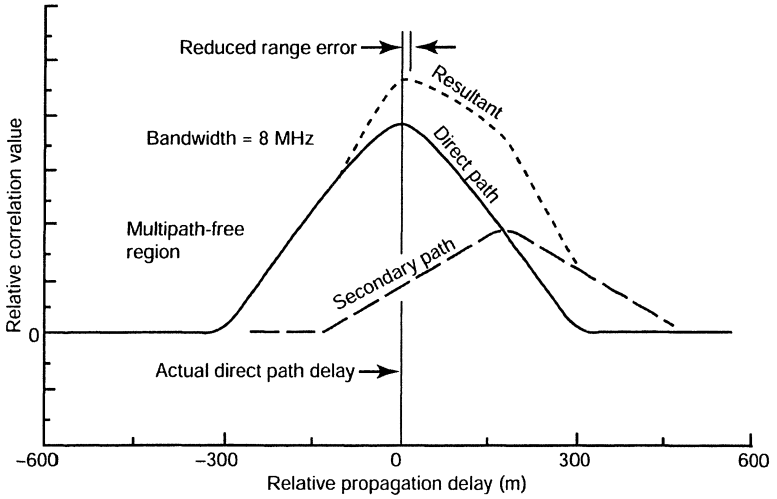


Fig. 7.3 Reduced multipath error with larger precorrelation bandwidth.

and without multipath. It is somewhat surprising that these facts were not recognized earlier by the GNSS community, given that they had been well known in radar circles for many decades.

A 2-MHz precorrelation bandwidth causes the peak of the direct-path cross-correlation function to be severely rounded, as illustrated in Fig. 7.2. Consequently, the sloping sides of a secondary-path component of the correlation function can significantly shift the location of the peak, as indicated in the figure. The result of using an 8-MHz bandwidth is shown in Fig. 7.3, where it can be noted that the sharper peak of the direct-path cross-correlation function is less easily shifted by the secondary-path component. It can also be shown that at larger bandwidths, the sharper peak is more resistant to disturbance by receiver thermal noise, even though the precorrelation signal-to-noise ratio is increased.

Another advantage of a larger precorrelation bandwidth is that the spacing between the early and late reference codes in a code tracking loop can be made smaller without significantly reducing the gain of the loop, hence the term narrow correlator. It can be shown that this causes the noises on the early and late correlator outputs to become more highly correlated, resulting in less noise on the loop error signal. An additional benefit is that the code tracking loop will be affected only by the multipath-induced distortions near the peak of the correlation function.

7.5.2.2 Leading-Edge Techniques Because the direct-path signal always precedes secondary-path signals, the leading (left-hand) portion of the correla-

tion function is uncontaminated by multipath, as is illustrated in Fig. 7.3. Therefore, if one could measure the location of just the leading part, it appears that the direct-path delay could be determined with no error due to multipath. Unfortunately, this seemingly happy state of affairs is illusory. With a small direct-/secondary-path separation, the uncontaminated portion of the correlation function is a minuscule piece at the extreme left, where the curve just begins to rise. In this region, not only is the signal-to-noise ratio relatively poor, but the slope of the curve is also relatively small, which severely degrades the accuracy of delay estimation.

For these reasons, the leading-edge approach best suits situations with a moderate to large direct-/secondary-path separation. However, even in these cases, there is the problem of making the delay measurement insensitive to the slope of the correlation function leading edge, which can vary with signal strength. Such a problem does not occur when measuring the location of the correlation function peak.

7.5.2.3 Correlation Function Shape-Based Methods Some GNSS receiver designers have attempted to determine the parameters of the multipath model from the shape of the correlation function. The idea has merit, but for best results, many correlations with different values of reference code delay are required to obtain a sampled version of the function shape. Another practical difficulty arises in attempting to map each measured shape into a corresponding direct-path delay estimate. Even in the simple two-path model (Eq. 7.14) there are six signal parameters, so that a very large number of correlation function shapes must be handled. An example of a heuristically developed shape-based approach called the early–late slope (ELS) method can be found in Ref. 13, while a method based on maximum-likelihood estimation (MLE) called the multipath-estimating delay-lock loop (MEDLL) is described in Ref. 14.

7.5.2.4 Modified Correlator Reference Waveforms Another new approach to multipath mitigation alters the waveform of the correlator reference PRN code to provide a cross-correlation function with inherent resistance to errors caused by multipath. Examples include the strobe correlator [15], the use of special code reference waveforms to narrow the correlation function developed in Refs. 16 and 17, and the gated correlator developed in Ref. 18 . These techniques take advantage of the fact that the range information in the received signal resides primarily in the chip transitions of the C/A-code. By using a correlator reference waveform that is not responsive to the flat portions of the C/A-code, the resulting correlation function can be narrowed down to the width of a chip transition, thereby being almost immune to multipath having a primary/secondary-path separation greater than 30–40 m. An example of such a reference waveform and the corresponding correlation function are shown in Fig. 7.4.

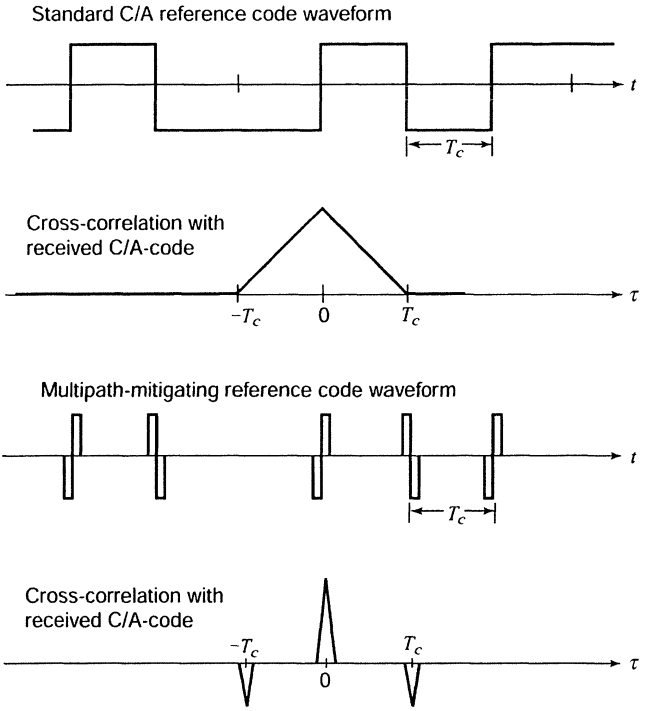


Fig. 7.4 Multipath-mitigating reference code waveform.

7.5.3 Multipath Mitigation Technology (MMT) Technology

Yet another approach to time-domain multipath mitigation is called MMT and incorporated a number of GNSS receivers manufactured by NovAtel Corporation of Canada. The MMT technique not only reaches theoretical performance limits described in Section 7.6 for both code and carrier phase ranging but also, compared to existing approaches, has the advantage that its performance improves as the signal observation time is lengthened. A description of MMT follows and also appears in a patent [19].

7.5.3.1 Description MMT is based on MLE. Although the theory of MLE is well-developed, its application to GNSS multipath mitigation has not been feasible until now due to the large amount of computation required. However, recent mathematical breakthroughs have solved this problem. Before introducing the MMT algorithm, we first briefly describe the process of MLE in the context of the multipath problem.

7.5.3.2 Maximum-Likelihood (ML) Multipath Estimation MLE is described in detail in Chapter 10. Its application to multipath mitigation is described below.

7.5.3.3 The Two-Path ML Estimator (MLE) The simplest ML estimator designed for multipath is based on a two-path model (one direct path and one secondary delayed path). For simplicity in describing MMT, we consider only this model, although generalization to additional paths is straightforward, and the MMT algorithm can be implemented for such cases. It is assumed that the received signal has been frequency-shifted to baseband, and the navigation data have been stripped off. The two-path signal model is

$$r(t) = A_1 e^{j\phi_1} m(t-\tau_1) + A_2 e^{j\phi_2} m(t-\tau_2) + n(t). \tag{7.24}$$

In this model, the parameters A_1, ϕ_1, τ_1, respectively, are the direct-path signal amplitude, phase, and delay, and the parameters A_2, ϕ_2, and τ_2 are the corresponding parameters for the secondary path. The code modulation is denoted by $m(t)$, and the noise function $n(t)$ is an additive zero-mean complex Gaussian noise process with a flat power spectral density. It will be convenient to group the multipath parameters into the vector

$$\bar{\theta} = [A_1, \phi_1, \tau_1 A_2, \phi_2, \tau_2]. \tag{7.25}$$

Observation of the received signal $r(t)$ is accomplished by sampling it on the time interval $[0, T]$ to produce a complex observed vector $\bar{r}$.

The ML estimate of the multipath parameters is the vector $\bar{\theta}$ of parameter values that maximizes the likelihood function $p(\bar{r}|\bar{\theta})$, which is the probability density of the received signal vector conditioned on the values of the multipath parameters. In this maximization, the vector $\bar{r}$ is held fixed at its observed value. Within the vector $\hat{\bar{\theta}}$, the estimates $\hat{\tau}_1$ and $\hat{\phi}_1$ of direct-path delay and carrier phase are normally the only ones of interest. However, the ML estimate of these parameters requires that the likelihood function $p(\bar{r}|\bar{\theta})$ be maximized over the six-dimensional (6D) space of *all* multipath parameters (components of 0). For this reason, the unwanted parameters are called *nuisance parameters*.

Since the natural logarithm is a strictly increasing function, the maximization of $p(\bar{r}|\bar{\theta})$ is equivalent to maximization of $L(\bar{r};\bar{\theta}) = \ln p(\bar{r}|\bar{\theta})$, which is called the log-likelihood function. The log-likelihood function is often simpler than the likelihood function itself, especially when the noise in the observations is additive and Gaussian. In our application this is the case.

Maximization of $L(\bar{r};\bar{\theta})$ by standard techniques is a daunting task. A brute-force approach is to find the maximum by a search over the 6D multipath parameter space, but it takes too long to be of practical value. Reliable gradient-based or hill climbing methods are too slow to be useful. Finding the maximum using differential calculus is difficult because of the nonlinearity of the resulting equations and the possibility of local maxima that are not global maxima. Iterative solution techniques are often difficult to analyze and may not converge to the correct solution in a timely manner, if they converge at

all. As we shall see, the MMT algorithm solves these problems by reducing the dimensionality of the search space.

7.5.3.4 Asymptotic Properties of ML Estimators MLE is used by MMT not only because it can be made computationally simple enough to be practical but also because ML estimators have desirable asymptotic properties (Asymptotic refers to the behavior of an estimator when the error becomes small. In GNSS, this occurs when E/N_0 is sufficiently large.):

1. The ML estimate of a parameter asymptotically converges in probability to the true parameter value.
2. The ML estimate is asymptotically efficient; that is, the ratio of the variance of the estimation error to the Cramer–Rao bound approaches unity.
3. The ML estimate is asymptotically Gaussian.

7.5.3.5 The MMT Multipath Mitigation Algorithm The MMT algorithm uses several mathematical techniques to solve what would otherwise be intractable computational problems. The first of these is a nonlinear transformation on the multipath parameter space to permit rapid computation of a log-likelihood function that has been partially maximized with respect to all of the multipath parameters except for the path delays. Thus, final maximization requires a search in only two dimensions for the two-path case, aided by acceleration techniques.

A new method of signal compression, described in Section 7.5.3.10, is used to transform the received signal into a very small vector on which MMT can operate very rapidly.

A major advantage of the MMT algorithm is that its performance improves with increasing E/N_0, the ratio of signal energy E to noise power spectral density. This is not true for most GNSS multipath mitigation methods because their estimation error is in the form of an irreducible bias. Additionally, the MMT algorithm provides ML estimates of all parameters in the multipath model and can utilize known bounds on the magnitudes of the secondary paths, if available, to improve performance.

7.5.3.6 The MMT Baseband Signal Model In the complex baseband signal $r(t)$ given by Eq. 7.24, it is assumed that the signal has been Doppler-compensated and stripped of the 50-bps navigation data modulation. In developing the MMT algorithm, it is useful to separate $r(t)$ into its real component, $x(t)$, and imaginary component, $y(t)$:

$$\left.\begin{aligned} x(t) &= A_1 \cos\phi_1 m(t-\tau_1) + A_2 \cos\phi_2 m(t-\tau_2) + n_x(t), \\ y(t) &= A_1 \sin\phi_1 m(t-\tau_1) + A_2 \sin\phi_2 m(t-\tau_2) + n_y(t), \end{aligned}\right] \tag{7.26}$$

where $n_x(t)$ and $n_y(t)$ are independent, real-valued, zero-mean Gaussian noise processes with flat power spectral density.

7.5.3.7 Baseband Signal Vectors The real and imaginary signal components are synchronously sampled on $[0, T]$ at the Nyquist rate $2W$, corresponding to the low-pass baseband bandwidth W, to produce the vectors

$$\left.\begin{aligned}\bar{x} &= (x_1, x_2, \ldots, x_M),\\ \bar{y} &= (y_1, y_2, \ldots, y_M),\end{aligned}\right] \tag{7.27}$$

in which the noise components of distinct samples are essentially uncorrelated (hence independent, since the noise is Gaussian).

7.5.3.8 The Log-Likelihood Function The ML estimates of the six parameters in the vector $\bar{\theta}$ given by (Eq. 7.25) are obtained by maximizing the log-likelihood function with respect to these parameters. For MMT, the log-likelihood function is

$$\begin{aligned}L(\bar{x}, \bar{y}|\bar{\theta}) &= \ln\left[p(\bar{x}, \bar{y}|\bar{\theta})\right]\\ &= \ln C_1\\ -C_2\sum_{k=1}^{M}&\left[\begin{matrix}x_k - A_1\cos\theta_1 m_k(\tau_1)\\ -A_2\cos\theta_2 m_k(\tau_2)\end{matrix}\right]^2\\ -C_2\sum_{k=1}^{M}&\left[\begin{matrix}y_k - A_1\cos\theta_1 m_k(\tau_1)\\ -A_2\cos\theta_2 m_k(\tau_2)\end{matrix}\right]^2,\end{aligned} \tag{7.28}$$

where

$$\begin{aligned}C_1 &= \left(\frac{1}{\sqrt{2\pi\sigma}}\right)^M\\ C_2 &= \frac{1}{2\sigma^2}\end{aligned} \tag{7.29}$$

σ^2 = noise variance of $x(t)$ and $y(t)$
$m_k(\tau_1)$ = kth sample of $m(t - \tau_1)$
$m_k(\tau_2)$ = kth sample of $m(t - \tau_2)$

Replacing the summations in (Eq. 7.28) by integrals and utilizing the fact that C_1 and $-C_2$ are negative constants that do not depend on the multipath parameters, maximization of Eq. 7.28 is equivalent to *minimization* of

$$\begin{aligned}\Gamma = &\int_0^T\left[\begin{matrix}x(t) - A_1\cos\phi_1 m(t-\tau_1)\\ -A_2\cos\phi_2 m_k(t-\tau_2)\end{matrix}\right]^2 dt\\ &+\int_0^T\left[\begin{matrix}y(t) - A_1\sin\phi_1 m(t-\tau_1)\\ -A_2\sin\phi_2 m_k(t-\tau_2)\end{matrix}\right]^2 dt\end{aligned} \tag{7.30}$$

with respect to the six multipath parameters. This is a highly coupled, nonlinear minimization problem on the 6D space spanned by the parameters A_1, ϕ_1, τ_1, A_2, ϕ_2, and τ_2. Standard minimization techniques such as a gradient search on this space or ad hoc iterative approaches are either unreliable or too slow to be useful.

However, a major breakthrough results by using the invertible transformation

$$\left.\begin{aligned} a &= A_1 \cos\phi_1 \quad c = A_1 \sin\phi_1 \\ b &= A_2 \cos\phi_2 \quad d = A_2 \sin\phi_2 \end{aligned}\right\}. \tag{7.31}$$

When this transformation is applied and the integrands in (Eq. 7.30) are expanded, the problem becomes one of minimizing

$$\begin{aligned} \Gamma = & \int_0^T \left[x^2(t) + y^2(t)\right] dt \\ & + \left(a^2 + b^2 + c^2 + d^2\right) R_{mm}(0) \\ & - 2aR_{xm}(\tau_1) - 2bR_{xm}(\tau_2) + 2abR_{mm}(\tau_1 - \tau_2) \\ & - 2cR_{ym}(\tau_1) - 2dR_{ym}(\tau_2) + 2cdR_{mm}(\tau_1 - \tau_2). \end{aligned} \tag{7.32}$$

Note that Γ in Eq. 7.32 is quadratic in a, b, c, and d, and uses the correlation functions

$$\left.\begin{aligned} R_{xm}(\tau) &= \int_0^T x(t)m(t-\tau)\,dt \\ R_{ym}(\tau) &= \int_0^T y(t)m(t-\tau)\,dt \\ R_{mm}(\tau) &= \int_0^T m(t)m(t-\tau)\,dt \end{aligned}\right\}. \tag{7.33}$$

Thus, minimization of Eq. 7.32 with respect to a, b, c, and d can be accomplished by taking partial derivatives with respect to these parameters, resulting in the linear system

$$\left.\begin{aligned} 0 &= \frac{\partial\Gamma}{\partial a} = 2aR_{mm}(0) - 2R_{xm}(\tau_1) + 2bR_{mm}(\tau_1 - \tau_2) \\ 0 &= \frac{\partial\Gamma}{\partial b} = 2bR_{mm}(0) - 2R_{xm}(\tau_2) + 2aR_{mm}(\tau_1 - \tau_2) \\ 0 &= \frac{\partial\Gamma}{\partial c} = 2cR_{mm}(0) - 2R_{ym}(\tau_1) + 2dR_{mm}(\tau_1 - \tau_2) \\ 0 &= \frac{\partial\Gamma}{\partial d} = 2dR_{mm}(0) - 2R_{ym}(\tau_2) + 2cR_{mm}(\tau_1 - \tau_2) \end{aligned}\right]. \tag{7.34}$$

For each pair of values of τ_1 and τ_2, this linear system can be explicitly solved for the minimizing values of a, b, c, and d. Thus, the space to be searched for a minimum of (7.32) (i.e., Eq. 7.32) is now 2D instead of 6D. The minimization procedure is as follows. Search the (τ_1, τ_2) domain. At each point (τ_1, τ_2), compute the values of the correlation functions in the system (Eq. 7.34) and then solve the system to find the values of a, b, c, and d that minimize Γ at that point. Identify the point $(\hat{\tau}_1, \hat{\tau}_2)_{\mathrm{ML}}$ where the smallest of all such minima is obtained, as well as the associated minimizing values of a, b, c, and d. Transform these values of a, b, c, and d back to the estimates $\hat{A}_{1\mathrm{ML}}$, $\hat{A}_{2\mathrm{ML}}$, $\hat{\phi}_{1\mathrm{ML}}$, $\hat{\phi}_{2\mathrm{ML}}$ by using the inverse of transformation (7.29), which is

$$\left.\begin{array}{ll} A_1 = \sqrt{a^2 + c^2} & A_2 = \sqrt{b^2 + d^2} \\ \phi_1 = \arctan 2(a, c) & \phi_2 = \arctan 2(b, d) \end{array}\right\}. \tag{7.35}$$

7.5.3.9 Secondary-Path Amplitude Constraint In the majority of multipath scenarios, the amplitudes of secondary-path signals are smaller than that of the direct path. The multipath mitigation performance of MMT can be significantly improved by minimizing Γ in Eq. 7.32 subject to the constraint

$$\frac{A_2}{A_1} \le \alpha, \tag{7.36}$$

where α is a positive constant (a typical value is 0.7). The constraint in terms of the transformed parameters a, b, c, and d is

$$b^2 + d^2 \le \alpha^2 \left(a^2 + c^2\right). \tag{7.37}$$

The constrained minimization of Eq. 7.32 uses the method of Lagrange multipliers.

7.5.3.10 Signal Compression In the MMT algorithm, the correlation functions $R_{xm}(\tau)$, $R_{ym}(\tau)$, and $R_{mm}(\tau)$ defined by Eq. 7.33 and appearing in Eq. 7.34 are computed very rapidly by first using a process called signal compression, in which the large number of signal samples (on the order of $10^8 - 10^9$) that would normally be involved is reduced to only a few tens of samples (the exact number depends on which type of GNSS signal is being processed). This processing is easily done in real time.

The correlation functions appearing in Eq. 7.33 have the form

$$R(\tau) = \int_0^T r(t) m(t - \tau) dt, \tag{7.38}$$

where $r(t)$ is a given function and $m(t)$ is a replica of the code modulation, which includes the effects of filtering in the satellite and receiver. The calculation of $R(\tau)$ in a conventional receiver is ordinarily not computationally

difficult because in such receivers, $m(t)$ can be an ideal chipping sequence with only the values ±1, and the multiplications of samples of the integrand of Eq. 7.36 then become trivial. Furthermore, conventional receivers track only the peak of the correlation function so that $R(\tau)$ needs to be computed for only a few values of τ (usually for early, punctual, and late correlations). However, the MMT algorithm cannot employ these simplifications. The function $m(t)$ used by MMT must include the aforementioned effects of filtering, thus requiring multibit multiplications (typically numbering in the millions) in the calculation of $R(\tau)$. Furthermore, $R(\tau)$ must be calculated for many values of τ to obtain high resolution for accurate estimation of direct-path delay in the presence of multipath.

These difficulties are circumvented by using signal compression. To simplify its description, we assume that the correlation function $R(\tau)$ in Eq. 7.36 is a cyclic correlation over one period T of the replica code $m(t)$, in which $m(t - \tau)$ is a rotation by τ (right for positive τ and left for negative τ). However, compression can be accomplished over an arbitrary interval of observation of the function $r(t)$, in which many periods of a received PN code occur, and furthermore, the correlation function need not be cyclic.

A single period of replica code can be written as

$$m(t) = \sum_{k=0}^{N-1} \varepsilon_k c(t - kT_c), \tag{7.39}$$

where T_c is the duration of each chip, ε_k is the chip polarity (either +1 or –1), and N is the number of chips in one period of the code. The function $c(t)$ is the response of the combined satellite and receiver filtering to a single ideal chip of the code. This ideal chip has a constant value of 1 on the interval $0 \le t \le T_c$. Because the filtering is linear and time invariant, it follows that $m(t)$ is the filter response to the entire code sequence. The index k identifies the individual chips of the code, where $k = 0$ identifies the epoch chip, defined as the first chip of the chipping sequence.

The compressed signal $\tilde{r}(t)$ is defined by

$$\tilde{r}(t) = \sum_{k=0}^{N-1} \varepsilon_k r(t + kT_c). \tag{7.40}$$

In this expression $\varepsilon_k r(t - kT_c)$ is $r(t)$ weighted by ε_k and left-rotated by kT_c. In GNSS applications, the compressed signal has the very nice property that essentially all of its energy (excluding noise) is concentrated into a pulse of one filtered chip in duration. This is made evident by noting that the received signal $r(t)$ without multipath can be expressed as

$$r(t) = am(t - \tau_0) + n(t) = a\left[\sum_{j=0}^{N-1} \varepsilon_j c(t - \tau_0 - jT_c)\right] + n(t), \tag{7.41}$$

where a is the signal amplitude, τ_0 is the signal delay, $n(t)$ is noise, and all time shifts are rotations (i.e., cyclic over one code period). Substitution of this expression into (Eq. 7.40) gives

$$\begin{aligned}\tilde{r}(t) &= \sum_{k=0}^{N-1} \varepsilon_k r(t+kT_c) \\ &= \sum_{k=0}^{N-1} \varepsilon_k \left\{ \left[\sum_{j=0}^{N-1} \varepsilon_j c(t-\tau_0+kT_c-jT_c) \right] + n(t+kT_c) \right\} \\ &= \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} \varepsilon_k \varepsilon_j c[t-\tau_0+(k-j)T_c] + \sum_{k=0}^{N-1} \varepsilon_k n(t+kT_c) \\ &= \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} \varepsilon_k \varepsilon_j c[t-\tau_0+(k-j)T_c] + \tilde{n}(t),\end{aligned} \tag{7.42}$$

where the double summation is the compressed signal component and the single summation is the compressed noise function $\tilde{n}(t)$. The terms in the double summation can be grouped into N groups such that each group contains N terms having the same value of $k-j$ modulo N. Thus, $\tilde{r}(t)$ will be the summation of N group sums plus $\tilde{n}(t)$. The group sum corresponding to particular value p of $k-j$ modulo N is $c[t-\tau_0+pT_c]$ weighted by the sum of terms $\varepsilon_j\varepsilon_k$, which satisfy $k-j=p$ modulo N. Since T_c is the duration of $c(t)$ before filtering, it can be seen that $\tilde{r}(t)$ consists of a concatenation of N weighted and translated copies of $c(t)$, which do not overlap, except for a trailing transient from each copy due to filtering.

7.5.3.11 Properties of the Compressed Signal If the number of chips N is sufficiently large (on the order of 10_3 or more), the autocorrelation function of the GNSS chipping sequence has the property that the group sums in which $k-j \neq 0$ modulo N are negligible compared to the group sum in which $k-j=0$ modulo N. Furthermore, the sum of all of these small group sums is also negligible because the translations of the weighted copies of $c(t)$ prevent the small group sums from accumulating to large values. Thus, to a very good approximation, the double summation in Eq. 7.42 is just the sum of the terms where $k-j=0$ modulo N:

$$\tilde{r}(t) \cong \left[\sum_{k=0}^{N-1} \varepsilon_k^2 c(t-\tau_0) \right] + \tilde{n}(t) = Nc(t-\tau_0) + \tilde{n}(t). \tag{7.43}$$

This is a very significant result because it tells us that the compressed received signal is essentially just the single weighted filtered chip $Nc(t-\tau_0)+\tilde{n}$ plus noise, with small "sidelobe" chips to either side. Furthermore, the compression process provides a processing gain of 10 log N dB. Since a receiver can measure the delay τ_0, a window can be constructed that need be long enough only to

contain $Nc(t - \tau_0)$, and the sidelobe chips as well as all noise outside this window can be rejected. The required length of the window is $T_c + \delta$, where δ is large enough to accommodate the measurement uncertainty of τ_0, the trailing transient due to filtering, and any multipath components with delays larger than τ_0 (almost certainly the only multipath components having significant amplitude are found within one chip of the direct-path delay). Thus, the window length is somewhat larger than the one-chip duration of the code, a quantity much smaller than the length T of the observed signal $r(t)$, which must include all N chips of the code. It is because of this result that $\tilde{r}(t)$ can justifiably be called a compressed signal. An illustration of the compressed signal is shown in Fig. 7.5.

If N is sufficiently large, the processing gain is great enough to make the compressed signal within the window visible with very little noise, so that small subtleties in the chip waveshape due to multipath or other causes can easily be seen. This property is very beneficial for signal integrity monitoring. It has been put to practical use in GNSS receivers sold by the NovAtel Corporation, which calls its implementation the *vision correlator*.

The compressed signal also enjoys a *linearity property*: If $r(t) = a_1 r_1(t) + a_2 r_2(t)$, then $\tilde{r}(t) = a_1 \tilde{r}_1(t) + a_2 \tilde{r}_2(t)$. The linearity property is essential for the MMT to properly process a multipath-corrupted signal.

7.5.3.12 The Compression Theorem Most importantly, the compressed signal can be used to drastically reduce the amount of computation of the correlation function $R(\tau)$ in Eq. 7.38. The basis for this assertion is the following theorem:

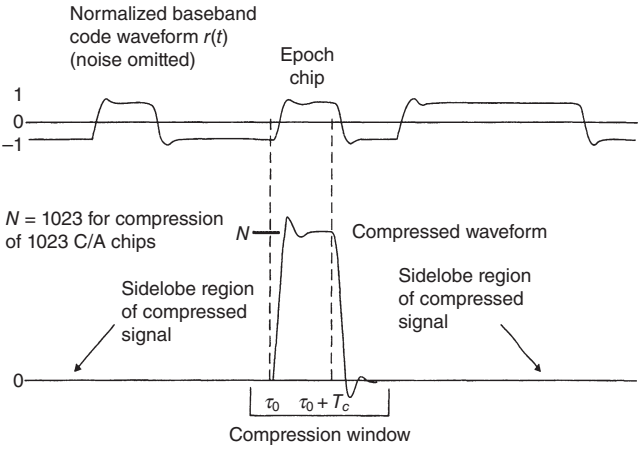


Fig. 7.5 Compression of the received signal.

The correlation function

$$R(\tau) = \int_0^T r(t)m(t-\tau)du, \tag{7.44}$$

can be computed by the alternate method

$$R(\tau) = \int_0^T \tilde{r}(t)c(t-\tau)du. \tag{7.45}$$

Proof:

$$\left.\begin{aligned} R(\tau) &= \int_0^T r(t)m(t-\tau)dt \\ &= \int_0^T r(t)\left[\sum_{k=0}^{N-1}\varepsilon_k c(t-kT_c-\tau)\right]dt \\ &= \sum_{k=0}^{N-1}\varepsilon_k \int_0^T r(t)c(t-kT_c-\tau)dt \\ &= \sum_{k=0}^{N-1}\varepsilon_k \int_0^T r(u+kT_c)c(u-\tau)du \quad (\text{using } u = t-kT_c) \\ &= \int_0^T \left[\sum_{k=0}^{N-1}\varepsilon_k r(u+kT_c)\right]c(u-\tau)du \\ &= \int_0^T \tilde{r}(u)c(u-\tau)du \end{aligned}\right\}. \tag{7.46}$$

This theorem shows that $R(\tau)$ can be computed by cross correlating the compressed signal $\tilde{r}(t)$ with the very short function $c(t)$. Furthermore, since we have already noted that the significant portion of $\tilde{r}(t)$ also spans a short time interval, the region surrounding the correlation peak of $R(\tau)$ can be obtained with far less computation than the original correlation (Eq. 7.44). The bottom line is that the cross correlations in Eq. 7.33 used by MMT can be calculated very efficiently by using the compressed versions of the signals $x(t)$, $y(t)$, and $m(t)$.

7.5.4 Performance of Time-Domain Methods

7.5.4.1 Ranging with the C/A-Code Typical C/A-code ranging performance curves for several multipath mitigation approaches are shown in Fig. 7.6 for the case of an in-phase secondary path with amplitude one-half that of the direct path. Even with the best available methods (other than MMT), peak

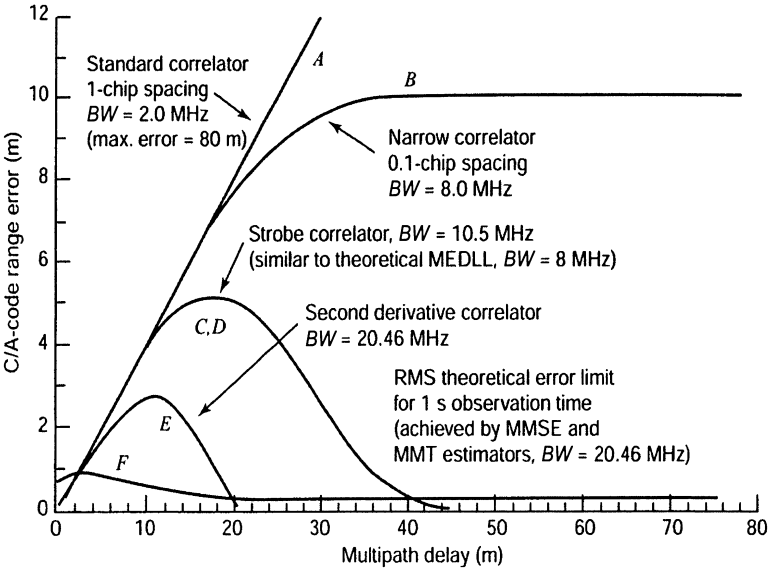


Fig. 7.6 Performance of various multipath mitigation approaches.

range errors of 3–6 m are not uncommon. It can be observed that the error tends to be largest for "close-in" multipath, where the separation of the two paths is on the order of 10 m. Indeed, this region poses the greatest challenge in multipath mitigation research because the extraction of direct-path delay from a signal with small direct/secondary-path separation is an ill-conditioned parameter estimation problem.

A serious limitation of most existing multipath mitigation algorithms is that the residual error is mostly in the form of a bias that cannot be removed by further filtering or averaging. On the other hand, the above mentioned MMT algorithm overcomes this limitation and also appears to have significantly better performance than other published algorithms, as is indicated by curve *F* of Fig. 7.6.

7.5.4.2 Carrier Phase Ranging The presence of multipath also causes errors in estimating carrier phase, which limits the performance in surveying and other precision applications, particularly with regard to carrier phase ambiguity resolution. Not all current multipath mitigation algorithms are capable of reducing multipath-induced phase error. The most difficult situation occurs at small separations between the direct and secondary paths (less than a few meters). It can be shown that, under such conditions, essentially no mitigation is theoretically possible. Typical phase error curves for the MMT algorithm, which appears to have the best performance of published methods, is shown in Fig. 7.7 [16].

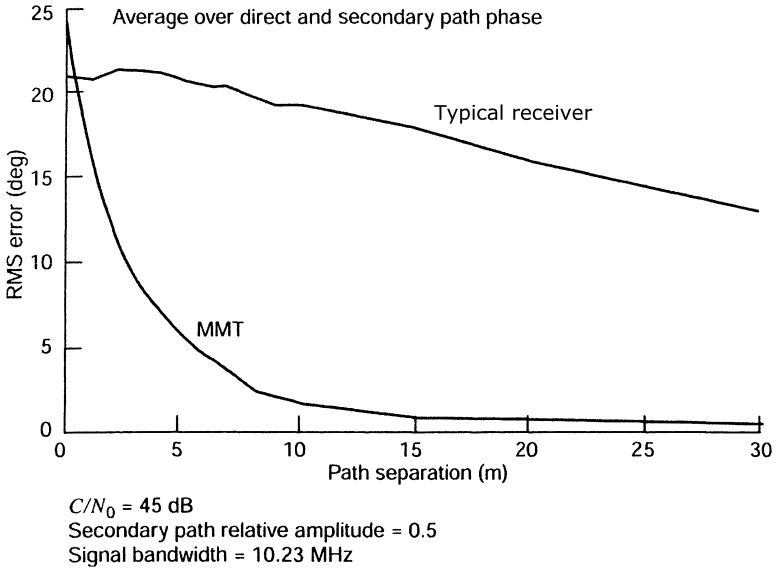


Fig. 7.7 Residual multipath phase error using MMT algorithm.

7.5.4.3 Testing Receiver Multipath Performance Conducting tests of receiver multipath mitigation performance on either an absolute or a comparative basis is often done in two ways. For very controlled and repeatable multipath error testing, an advanced GNSS simulator is used to represent an operational scenario whereby multiple signals (direct) and multipath are simulated and fed into the GNSS receiver. The results can then be compared to the known simulated measurements and position of the user GNSS receiver.

Another way to perform multipath performance analysis is in the real operational environment of the receiver. This type of testing will evaluate the entire system and processing performed, i.e., antenna, receiver correlation, postcorrelation processing. An analysis technique commonly referred to as a code-minus-carrier (CMC) technique is used. This technique is most often applied to analyze the code multipath in a postprocessing fashion whereby the code measurement is detrended by the carrier phase measurement and all other error sources are removed, except for the multipath to be analyzed. The predicted multipath error can then be used to analyze the performance of the GNSS receiver systems or compared with other prediction methods for validation.

7.6 THEORETICAL LIMITS FOR MULTIPATH MITIGATION

7.6.1 Estimation-Theoretic Methods

Relatively little has been published on multipath mitigation from the fundamental viewpoint of statistical estimation theory despite the power of its

methods and its ability to reach theoretical performance limits in many cases. Knowledge of such limits provides a valuable benchmark in receiver design by permitting an accurate assessment of the potential payoff in developing techniques that are better than those in current use. Of equal importance is the revelation of the signal processing operations that can reach performance bounds. Although it may not be feasible to implement the processing directly, its revelation often leads to a practical method that achieves nearly the same performance.

7.6.1.1 Optimality Criteria In discussing theoretical performance limits, it is important to define the criterion of optimality. In GPS, the optimal range estimator is traditionally considered to be the minimum-variance unbiased estimator (MVUE), which can be realized by properly designed receivers. However, in Ref. 20, it is shown that the standard deviation of a MVUE designed for multipath becomes infinite as the primary-to-secondary-path separation approaches zero. For this reason, it seems that a better criterion of optimality would be the minimum root mean square RMS error, which can include both random and bias components. Unfortunately, it can be shown that no estimator exists having minimum RMS error for every combination of true multipath parameters.

7.6.2 Minimum Mean-Squared Error (MMSE) Estimator

There is an estimator that can be claimed optimal in a weaker sense. The MMSE estimator has the property that no other estimator has a uniformly smaller RMS error. In other words, if some other estimator has smaller RMS error than the MMSE estimator for some set of true multipath parameter values, then that estimator must have a larger RMS error than the MMSE estimator for some other set of values.

The MMSE estimator also has an important advantage not possessed by most current multipath mitigation methods in that the RMS error decreases as the length of the signal observation interval is increased.

7.6.3 Multipath Modeling Errors

Although a properly designed estimation-theoretic approach such as the MMSE estimator will generally outperform other methods, the design of such estimators requires a mathematical model of the multipath-contaminated signal containing parameters to be estimated. If the actual signal departs from the assumed model, performance degradation can occur. For example, if the model contains only two signal propagation paths but in reality the signal is arriving via three or more paths, large bias errors in range estimation can result. On the other hand, poorer performance (usually in the form of random error cause by noise) can also occur if the model has too many degrees of freedom. Striking the right balance in the number of parameters in the model

can be difficult if little information exists about the multipath reflection geometry.

7.7 EPHEMERIS DATA ERRORS

Small errors in the ephemeris data transmitted by each satellite cause corresponding errors in the computed position of the satellite (here we exclude the ephemeris error component of SA, which is regarded as a separate error source). Satellite ephemerides are determined by the master control station of the GNSS ground segment based on monitoring of individual signals by four monitoring stations. Because the locations of these stations are known precisely, an "inverted" positioning process can calculate the orbital parameters of the satellites as if they were users. This process is aided by precision clocks at the monitoring stations and by tracking over long periods of time with optimal filter processing. Based on the orbital parameter estimates thus obtained, the master control station uploads the ephemeris data to each satellite, which then transmits the data to users via the navigation data message. Errors in satellite position when calculated from the ephemeris data typically result in range errors on the order of 1–2 m. Improvements in satellite tracking will undoubtedly reduce this error further.

7.8 ONBOARD CLOCK ERRORS

Timing of the signal transmission from each satellite is directly controlled by its own atomic clock without any corrections applied. This time frame is called *space vehicle* (SV) *time*. A schematic of a rubidium atomic clock is shown in Fig. 7.8. Although the atomic clocks in the satellites are highly accurate, errors can be large enough to require correction. Correction is needed partly because it would be difficult to directly synchronize the clocks closely in all the satellites. Instead, the clocks are allowed some degree of relative drift that is estimated by ground station observations and is used to generate clock correction data in the GNSS navigation message. When SV time is corrected using this data, the result is called GNSS *time*. The time of transmission used in calculating pseudoranges must be GNSS time, which is common to all satellites.

The onboard clock error is typically less than 1 ms and varies slowly. This permits the correction to be specified by a quadratic polynomial in time whose coefficients are transmitted in the navigation message. The correction has the form

$$\Delta t_{\mathrm{sv}} = a_{f0} + a_{f1}(t_{\mathrm{sv}} - t_{oc}) + a_{f2}(t_{\mathrm{sv}} - t_{oc})^2 + \Delta t_{\mathrm{r}}, \tag{7.47}$$

with

$$t_{\mathrm{GNSS}} = t_{\mathrm{sv}} - \Delta t_{\mathrm{sv}}, \tag{7.48}$$

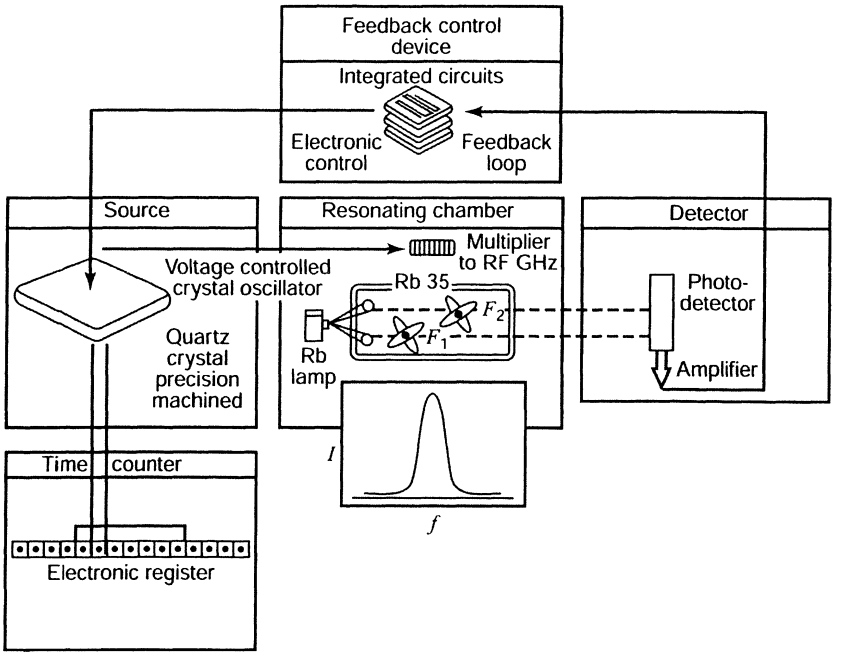


Fig. 7.8 Schematic of a rubidium atomic clock. RF, radio frequency.

where a_{f0}, a_{f1}, a_{f2} are the correction coefficients, t_{sv} is SV time, and Δt_r is a small relativistic clock correction caused by the orbital eccentricity. The clock data reference time t_{oc}, in seconds is broadcast in the navigation data message. The stability of the atomic clocks permits the polynomial correction given by Eq. 7.47 to be valid over a time interval of 4–6 h. After the correction has been applied, the residual error in GNSS time is typically less than a few nanoseconds, or about 1 m in range. Complete calculations of GNSS time are given as exercises in Chapter 4, Section 4.1.3.4.

7.9 RECEIVER CLOCK ERRORS

Because the navigation solution includes a solution for receiver clock error, the requirements for accuracy of receiver clocks is far less stringent than for GNSS satellite clocks. In fact, for receiver clocks, short-term stability over the pseudorange measurement period is usually more important than absolute frequency accuracy. In almost all cases, such clocks are quartz crystal oscillators with absolute accuracies in the 1- to 10-ppm range over typical operating temperature ranges. When properly designed, such oscillators typically have stabilities of 0.01–0.05 ppm over a period of a few seconds.

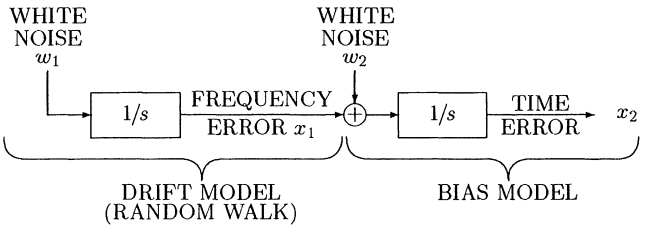


Fig. 7.9 Crystal clock error model.

Receivers that incorporate receiver clock error in the Kalman filter state vector need a suitable mathematical model of the crystal clock error. A typical model in the continuous-time domain is shown in Fig. 7.9, which is easily changed to a discrete version for the Kalman filter. In this model, the clock error consists of a bias (frequency) component and a drift (time) component. The frequency error component is modeled as a random walk produced by integrated white noise. The time error component is modeled as the integral of the frequency error after additional white noise (statistically independent from that causing the frequency error) has been added to the latter. In the model, the key parameters that need to be specified are the power spectral densities of the two noise sources, which depend on characteristics of the specific crystal oscillator used.

The continuous-time model has the form

$$\dot{x}_1 = w_1, \tag{7.49}$$

$$\dot{x}_2 = x_1 + w_2, \tag{7.50}$$

where $w_1(t)$ and $w_2(t)$ are independent zero-mean white-noise processes with known variances.

The equivalent discrete-time model has the state vector

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \tag{7.51}$$

and the stochastic sequence model

$$\mathbf{x}_k = \begin{bmatrix} 1 & 0 \\ \Delta t & 1 \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} w_{1,k-1} \\ w_{2,k-1} \end{bmatrix}, \tag{7.52}$$

where Δt is the discrete-time step and $\{w_{1,k-1}\}$, $\{w_{2,k-1}\}$ are independent zero-mean white-noise sequences with known variances.

7.10 SA ERRORS

Prior to May 1, 2000, SA was a mechanism adopted by the Department of Defense (DOD) to control the achievable navigation accuracy by nonmilitary GPS receivers. In the GPS SPS mode, the SA errors were specified to degrade navigation solution accuracy to 100 m (2D RMS) horizontally and 156 m (RMS) vertically. On May 1, 2000, the President of the United States announced the decision to discontinue this intentional degradation of GPS signals available to the public. The decision to discontinue SA was coupled with continuing efforts to upgrade the military utility of systems using GPS and was supported by threat assessments that concluded that setting SA to zero would have minimal impact on U.S. national security. The decision was part of an ongoing effort to make GPS more responsive to civil and commercial users worldwide.

7.11 ERROR BUDGETS

For purposes of analyzing the effects of the errors discussed above, it is convenient to convert each error into an equivalent range error experienced by a user, which is called the *user-equivalent range error* (UERE). In general, the errors from different sources will have different statistical properties. For example, satellite clock and ephemeris errors tend to vary slowly with time and appear as biases over moderately long time intervals, perhaps hours. On the other hand, errors due to receiver noise and quantization effects may vary much more rapidly, perhaps within seconds. Nonetheless, if sufficiently long time durations over many navigation scenarios are considered, all errors can be considered as zero-mean random processes that can be combined to form a single UERE. This is accomplished by forming the root sum square (RSS) of the UERE errors from all sources:

$$\text{UERE} = \sqrt{\sum_{i=1}^{n} (\text{UERE})_i^2}\,. \tag{7.53}$$

Figure 7.10 depicts the various GPS UERE errors and their combined effect for both C/A-code and P(Y)-code navigation at the $1 - \sigma$ level.

The UERE for the C/A-code user is typically about 19 m. It can be seen that, for such a user, the dominant error sources in nondifferential operations are multipath, receiver noise/resolution, and ionospheric and tropospheric delay (however, recent advances in receiver technology have in some cases significantly reduced receiver noise/resolution errors). On the other hand, the P(Y)-code user has a significantly smaller UERE of about 6 m for the following reasons:

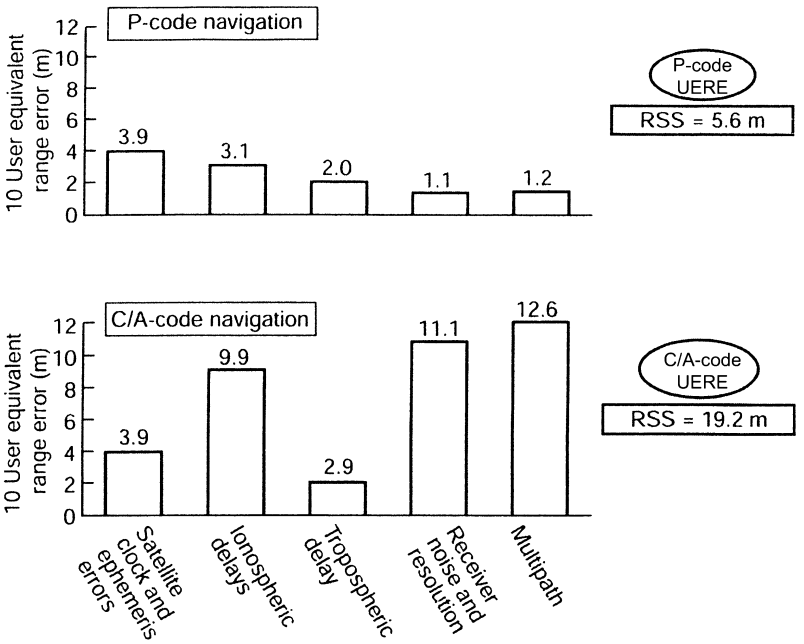


Fig. 7.10 GPS UERE budget.

1. The full use of the L1 and L2 signals permits significant reduction of ionospheric error.
2. The wider bandwidth of the P(Y)-codes greatly reduces errors due to multipath and receiver noise.

PROBLEMS

7.1 Using the values provided for Klobuchar's model in Section 7.2.1, calculate the ionospheric delay and plot the results.

7.2 Assume that a direct-path GNSS L1 C/A-code signal arrives with a phase such that all of the signal power lies in the baseband *I* channel, so that the baseband signal is purely real. Further assume an infinite signal bandwidth so that the cross correlation of the baseband signal with an ideal C/A reference code waveform will be an isosceles triangle 600 m wide at the base.

(a) Suppose that in addition to the direct-path signal there is a secondary-path signal arriving with a relative time delay of precisely 250 L1 carrier cycles (so that it is in phase with the direct-path signal) and with an amplitude one-half that of the direct path. Calculate the pseudorange error that would result, including its sign, under noiseless

conditions. Assume that pseudorange is measured with a delay-lock loop using 0.1-chip spacing between the early and late reference codes. (*Hint*: The resulting cross-correlation function is the superposition of the cross-correlation functions of the direct- and secondary-path signals.)

(b) Repeat the calculations of part (a) but with a secondary-path relative time delay of precisely 250.5 carrier cycles. Note that in this case, the secondary-path phase is 180° out of phase with the direct-path signal but still lies entirely in the baseband *I* channel.

7.3 (a) Using the discrete matrix version of the receiver clock model given by Eq. 7.52, find the standard deviation σ_{w1} of the white-noise sequence $w_{1,k}$ needed in the model to produce a frequency standard deviation σ_{x1} of 1 Hz after 10 min of continuous oscillator operation. Assume that the initial frequency error at $t = 0$ is zero and that the discrete-time step Δt is 1 s.

(b) Using the assumptions and the value of σ_{w1} found in part (a), find the standard deviation σ_{x2} of the bias error after 10 min. Assume that $\sigma_{w2} = 0$

(c) Show that σ_{x1} and σ_{x2} approach infinity as the time t approaches infinity. Will this cause any problems in the development of a Kalman filter that includes estimates of the clock frequency and bias error?

7.4 The peak electron density in the ionosphere occurs in a height range of

(a) 50–100 km

(b) 250–400 km

(c) 500–700 km

(d) 800–1000 km.

7.5 The refractive index of the gaseous mass in the troposphere is

(a) Slightly higher than unity

(b) Slightly lower than unity

(c) Unity

(d) Zero.

7.6 If the range measurements for two simultaneously tracking satellites in a receiver are differenced, then the differenced measurement will be free of

(a) receiver clock error only

(b) satellite clock error and orbital error only

(c) ionospheric delay error and tropospheric delay error only

(d) ionospheric delay error, tropospheric delay error, satellite clock error, and orbital error only.

7.7 Zero baseline test (code) can be performed to estimate

(a) receiver noise and multipath

(b) receiver noise

(c) receiver noise, multipath, and atmospheric delay errors

(d) none of the above.

7.8 What are the purposes of SA and antispoofing (AS)?

7.9 Derive the multipath formula equivalent to Eq. 7.23 for L2 using the same notation as in Eq. 7.17.

7.10 Calculate the ionospheric delay using dual-frequency carrier phases.

REFERENCES

[1] W. A. Feess and S. G. Stephens, "Evaluation of GPS Ionospheric Time Delay Algorithm for Single Frequency Users," *Proceedings of the IEEE Position, Location, and Navigation Symposium (PLANS '86)*, Las Vegas, NV, November 4–7, 1986, New York, 1986, pp. 206–213.

[2] J. A. Klobuchar, "Ionospheric Time Delay Corrections for Advanced Satellite Ranging Systems," *NATO AGARD Conference Proceedings 209*, in *Propagation Limitations of Navigation and Positioning Systems*, NATO AGARD, Paris, 1976.

[3] RTCA, *Minimum Operational Performance Standards for Global Positioning System, Wide Area Augmentation System*, Document RTCA/D0-229, Radio Technical Commission for Aeronautics, Washington, DC, 1999.

[4] A. Mannucci, B. Wilson, and C. D. Edwards, "A New Method for Monitoring the Earth's Total Electron Content Using the GPS Global Network," *Proceedings of ION GPS-93*, Salt Lake City, UT, September 1993, pp. 22–24.

[5] M. S. Grewal, "Space-Based Augmentation for Global Navigation Satellite Systems," *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control* **59**(3): 497–504 (March 2012).

[6] M. B. El-Arini, R. S. Conker, T. W. Albertson, J. K. Reagan, J. A. Klobuchar, and P. H. Doherty, "Comparison of Real-Time Ionospheric Algorithms for a GPS Wide-Area Augmentation System," *Navigation, Journal of the Institute of Navigation* **41**(4), 393–413 (Winter 1994/1995).

[7] R. Moreno and N. Suard, "Ionospheric Delay Using Only LI: Validation and Application to GPS Receiver Calibration and to Inter-Frequency Bias Estimation," *Proceedings of The Institute of Navigation (ION)*, January 25–27, 1999, ION, Alexandria, VA, 1999, pp. 119–129.

[8] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *GPS: Theory and Practice*. Springer-Verlag, Vienna, 1997.

[9] H. W. Janes, R. B. Langley, and S. Newby, "Analysis of Tropospheric Delay Prediction Models: Comparisons with Ray-Tracing and Implications for GPS Relative Positioning," *Bulletin Geodisique* **65**(3), 151–161 (1991).

[10] G. Seeber, *Satellite Geodesy: Foundation, Methods, and Applications*. Walter de Gruyter, Berlin, 1993.

[11] L. Hagerman, Effects of Multipath on Coherent and Noncoherent PRN Ranging Receiver, Aerospace Report TOR-0073(3020-03)-3, Aerospace Corporation, Development Planning Division, El Segundo, CA, May 15, 1973.

[12] A. J. Van Dierendonck, P. Fenton, and T. Ford, "Theory and Performance of Narrow Correlator Spacing in a GPS Receiver," *Proceedings of the National Technical Meeting*, Institute of Navigation, San Diego, CA, 1992, pp. 115–124.

[13] B. Townsend and P. Fenton, "A Practical Approach to the Reduction of Pseudorange Multipath Errors in a L1 GPS Receiver," *Proceedings of ION GPS-94, 7th International Technical Meeting of the Satellite Division of the Institute of Navigation* (Salt Lake City, UT), Alexandria, VA, 1994, pp. 143–148.

[14] B. Townsend, D. J. R. Van Nee, P. Fenton, and K. Van Dierendonck, "Performance Evaluation of the Multipath Estimating Delay Lock Loop," *Proceedings of the National Technical Meeting*, Institute of Navigation, Anaheim, CA, 1995, pp. 277–283.

[15] L. Garin, F. van Diggelen, and J. Rousseau, "Strobe and Edge Correlator Multipath Mitigation for Code," *Proceedings of ION GPS-96, 9th International Technical Meeting of the Satellite Division of the Institute of Navigation* (Kansas City, MO), ION, Alexandria, VA, 1996, pp. 657–664.

[16] L. Weill, "GPS Multipath Mitigation by Means of Correlator Reference Waveform Design," *Proceedings of the National Technical Meeting*, Institute of Navigation (Santa Monica, CA), ION, Alexandria, VA, January 1997, pp. 197–206.

[17] L. Weill, "Application of Superresolution Concepts to the GPS Multipath Mitigation Problem," *Proceedings of the National Technical Meeting*, Institute of Navigation (Long Beach, CA), ION, Alexandria, VA, 1998, pp. 673–682.

[18] G. McGraw and M. Braasch, "GNSS Multipath Mitigation Using Gated and High Resolution Correlator Concepts," *Proceedings of the 1999 National Technical Meeting and 19th Biennial Guidance Test Symposium*, Institute of Navigation, San Diego, CA, 1999, pp. 333–342.

[19] L. Weill and B. Fisher, *Method for Mitigating Multipath Effects in Radio Ranging Systems*, U.S. Patent 6,031,881, February 29, 2000.

[20] L. Weill, "Achieving Theoretical Accuracy Limits for Pseudoranging in the Presence of Multipath," *Proceedings of ION GPS-95, 8th International Technical Meeting of the Satellite Division of the Institute of Navigation* (Palm Springs, CA), ION, Alexandria, VA, 1995, pp. 1521–1530.

[21] J. Dickman, C. Bartone, Y. Zhang, B. Thornburg, "Characterization and Performance of a Prototype Wideband Airport Pseudolite Multipath Limiting Antenna for the Local Area Augmentation System," *Proceedings of the 2003 National Technical Meeting of The Institute of Navigation*, Anaheim, CA, January 2003, pp. 783–793.

8

DIFFERENTIAL GNSS

8.1 INTRODUCTION

Differential global navigation satellite system (differential GNSS [DGNSS]) is a technique for reducing the error in GNSS-derived positions by using additional data from a reference GNSS receiver at a known location. The most common form of DGNSS involves determining the combined effects of navigation message ephemeris and satellite clock errors (including the effects of propagation) at a reference station and transmitting corrections, in real time, to a user's receiver. The receiver applies the corrections in the process of determining its position [1]. These include corrections for satellite ephemeris, clock errors, and atmospheric delay errors. Still other error sources cannot be corrected with DGNSS, that is, multipath errors and user receiver errors.

While there are various ways to implement DGNSS, most can be categorized as correction-based DGNSS and measurement/relative-based DGNSS. Correction-based DGNSS typically involves a reference/monitor station that is most often fixed and surveyed, whereby pseudorange corrections are generated with respect to the "true range" from the reference station to the space vehicle (SV). When all of the error terms are put into a single correction, per SV, and matched with the ephemeris set, it is often referred to as a lumped pseudorange correction. Some correction-based DGNSS architectures decompose the error sources and provide corrections for specific error terms (e.g., orbit, ionosphere). In measurement/relative-based DGNSS architectures, the emphasis is to send monitor/reference station measurements to the rover so that the user can perform difference processing and solve for the relative range vector (i.e., baseline) between the reference station and the mobile user.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

8.2 DESCRIPTIONS OF LOCAL-AREA DIFFERENTIAL GNSS (LADGNSS), WIDE-AREA DIFFERENTIAL GNSS (WADGNSS), AND SPACE-BASED AUGMENTATION SYSTEM (SBAS)

8.2.1 LADGNSS

LADGNSS is a form of differential GNSS (DGNSS) in which the user's GNSS receiver receives real-time pseudorange and, possibly, carrier phase corrections from a reference receiver generally located within the line of sight (LOS). The corrections account for the combined effects of navigation message ephemeris and satellite clock errors (including the effects of SA) and, usually, atmospheric propagation delay errors at the reference station. With the assumption that these errors are also common to the measurements made by the user's receiver, the application of the corrections will result in more accurate position solutions. [2].

8.2.2 WADGNSS

WADGNSS is a form of DGNSS in which the user's GNSS receiver receives corrections determined from a network of reference stations distributed over a wide geographic area. Separate corrections are usually determined for specific error sources, such as satellite clock, ionospheric propagation delay, and ephemeris. The corrections are applied in the user's receiver or attached computer in computing the receiver's position solutions. The corrections are typically supplied in real time by way of a geostationary communications satellite or through a network of ground-based transmitters. Corrections may also be provided at a later date for postprocessing collected data [2].

8.2.3 SBAS

8.2.3.1 Wide-Area Augmentation System (WAAS) WAAS enhances the GPS SPS and is available over a wide geographic area. The WAAS, developed by the Federal Aviation Administration (FAA) together with other agencies, provides WADGPS corrections, additional ranging signals from geostationary (geostationary earth orbit [GEO]) satellites, and integrity data on the GPS and GEO satellites [2]. Improvements to WAAS are still being made by Raytheon under a contract from the FAA. This section includes the latest improvements to WAAS with two new GEOs owned by the FAA and a third GEO leased from Inmarsat [3,4].

The FAA is currently in Phase III of WAAS. Significant technical modifications include development of an engineering model reference receiver that is L1C, L2C, and L5 capable, improved WAAS processing for ionospheric estimation, scintillation robustness, and signal quality (evil waveform) monitoring, and incorporation of additional geostationary satellites and associated ground facilities. WAAS is also preparing for the start of Phase IV—Dual Frequency

Operations. The FAA intends to replace the use of the GPS L2 P(Y) semicodeless signal with the use of the GPS L5 civil signal since Department of Defense (DOD) support of GPS L2 codeless/semicodeless capability is likely to be discontinued in 2020. In addition to the WAAS ground system transition from L2 to the L5 civil signal in WAAS Phase IV, the FAA also plans to introduce a new dual frequency SBAS navigation service, while retaining legacy single-frequency user services.

Each GEO uplink subsystem (GUS) includes a closed-loop control algorithm and special signal generator (SigGen) hardware. These ensure that the downlink signal to the users is controlled adequately to be used as a ranging source to supplement the GPS satellites in view.

The primary mission of WAAS is to provide a means for air navigation for all phases of flight in the National Airspace System (NAS) from departure, en route, arrival, and through approach. GPS augmented by WAAS offers the capability for both nonprecision approach (NPA) and precision approach (PA) within a specific service volume. A secondary mission of the WAAS is to provide a WAAS network time (WNT) offset between the WNT and Coordinated Universal Time (UTC) for non-navigation users.

WAAS provides improved en route navigation and PA capability to WAAS-certified avionics. The safety critical WAAS system consists of the equipment and software necessary to augment the DOD-provided GPS SPS. WAAS provides a signal in space (SIS) to WAAS-certified aircraft avionics using the WAAS for any FAA-approved phase of flight. The SIS provides two services: (1) data on GPS and GEO satellites and (2) a ranging capability.

The GPS satellite data are received and processed at widely dispersed wide-area reference stations (WRSs), which are strategically located to provide coverage over the required WAAS service volume. Data are forwarded to wide-area master stations (WMSs), which process the data from multiple WRSs to determine the integrity, differential corrections, and residual errors for each monitored satellite and for each predetermined ionospheric grid point (IGP). Multiple WMSs are provided to eliminate single-point failures within the WAAS network. Information from all WMSs is sent to each GUS and uplinked along with the GEO navigation message to GEO satellites. The GEO satellites downlink these data to the users via the GPS SPS L-band ranging signal (L1) frequency with GPS-type modulation. Each ground-based station/subsystem communicates via a terrestrial communications subsystem (TCS) (see Fig. 8.1).

In addition to providing augmented GPS data to the users, WAAS verifies its own integrity and takes any necessary action to ensure that the system meets the WAAS performance requirements. WAAS also has a system operation and maintenance function that provides status and related maintenance information to FAA airway facilities (AFs) NAS personnel.

Correction and verification (C&V) processes data from all WRSs to determine integrity, differential corrections, satellite orbits, and residual error bounds for each monitored satellite. It also determines ionospheric vertical

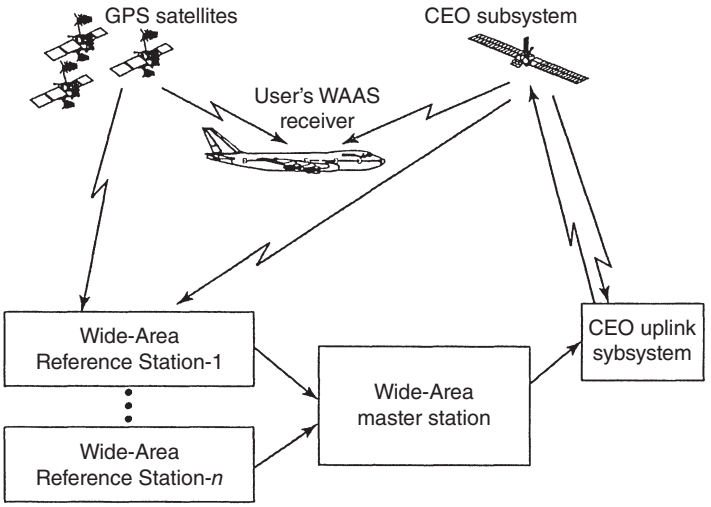


Fig. 8.1 WAAS top-level view.

delays and their residual error bounds at each of the IGPs. C&V schedules and formats WAAS messages and forwards them to the GUSs for broadcast to the GEO satellites.

C&V's capabilities are as follows:

1. Control C&V operations and maintenance (COM) supports the transfer of files, performs remotely initiated software configuration checks, and accepts requests to start and stop execution of the C&V application software.
2. Control C&V modes manage mode transitions in the C&V subsystem while the application software is running.
3. Monitor C&V (MCV) reports line replaceable unit (LRU) faults and configuration status. In addition, it monitors software processes and provides performance data for the local C&V subsystems.
4. Process input data (PID) selects and monitors data from the wide-area reference equipment (WREs). Data that pass PID screening are repackaged for other C&V capabilities. PID performs clock and L1 GPS Precise Positioning Service L-band ranging signal (L2) receiver bias calculations, cycle slip detection, outlier detection, data smoothing, and data monitoring. In addition, PID calculates and applies the windup correction to the carrier phase, accumulates data to estimate the pseudorange to carrier phase bias, and computes the ionosphere corrected carrier phase and measured slant delay.

5. Satellite Orbit Determination (SOD) determines the GPS and GEO satellite orbits and clock offsets, WRE receiver clock offsets, and troposphere delay.
6. Ionosphere Correction Computation (ICC) determines the L1 IGP vertical delays, grid ionosphere vertical error (GIVE) for all defined IGPs, and L1–L2 interfrequency bias for each satellite transmitter and each WRS receiver.
7. Satellite Correction Processing (SCP) determines the fast and long-term satellite corrections, including the user differential range error (UDRE). It determines the WNT and the GEO and WNT clock steering commands [3].
8. Independent Data Verification (IDV) compares satellite corrections, GEO navigation data, and ionospheric corrections from two independent computational sources, and if the comparisons are within limits, one source is selected from which to build the WAAS messages. If the comparisons are not within limits, various responses may occur, depending on the data being compared, all the way from alarms being generated to the C&V being faulted.
9. Message Output Processing (MOP) transmits messages containing independently verified results of C&V calculations to the GUS processing (GP) for broadcast.
10. C&V Playback (PLB) processes the playback data that have been recorded by the other C&V capabilities.
11. Integrity Data Monitoring (IDM) checks both the broadcast and the to-be-broadcast UDREs and GIVEs to ensure that they are properly bounding their errors. In addition, it monitors and validates that the broadcast messages are sent correctly. It also performs the WAAS time-to-alarm validation [5, 6].

WRS Algorithms Each WRS collects raw pseudorange (PR) and accumulated Doppler range (ADR) measurements from GPS and GEO satellites selected for tracking. Each WRS performs smoothing on the measurements and corrects for atmospheric effects, that is, ionospheric and tropospheric delays. These smoothed and atmospherically corrected measurements are provided to the WMS.

WMS Foreground (Fast) Algorithms The WMS foreground algorithms are applicable to real-time processing functions, specifically the computation of fast correction, determination of satellite integrity status, and WAAS message formatting. This processing is done at a 1-Hz rate.

WMS Background (Slow) Algorithms The WMS background processing consists of algorithms that estimate slowly varying parameters. These algorithms consist of WRS clock error estimation, grid ionospecific delay computation,

broadcast ephemeris computation, SOD, satellite ephemeris error computation, and satellite visibility computation.

IDV and Validation Algorithms This includes a set of WRS and at least one WMS, which enable monitoring the integrity status of GPS and the determination of wide-area DGPS correction data. Each WRS has three dual-frequency GPS receivers to provide parallel sets of measurement data. The presence of parallel data streams enables independent data verification and validation (IDV&V) to be employed to ensure the integrity of GPS data and their corrections in the WAAS messages broadcast via one or more GEOs. With IDV&V active, the WMS applies the corrections computed from one stream to the data from the other stream to provide verification of the corrections prior to transmission. The primary data stream is also used for the validation phase to check the active (already broadcast) correction and to monitor their SIS performance. These algorithms are continually being improved [5, 7–11].

8.2.3.2 European Global Navigation Overlay System (EGNOS) EGNOS is a joint project of the European Space Agency, the European Commission, and the European Organization for the Safety of Air Navigation (Eurocontrol). Its primary service area is the European Civil Aviation Conference (ECAC) region. However, several extensions of its service area to adjacent and more remote areas are under study. An overview of the EGNOS system architecture is presented in Fig. 8.2, where

[RIMS] are the Ranging and Integrity Monitoring Stations
[MCC] is the Mission and Control Center
[NLES] are the Navigation Land Earth Stations.
[PACF] is the Performance Assessment and Checkout Facility

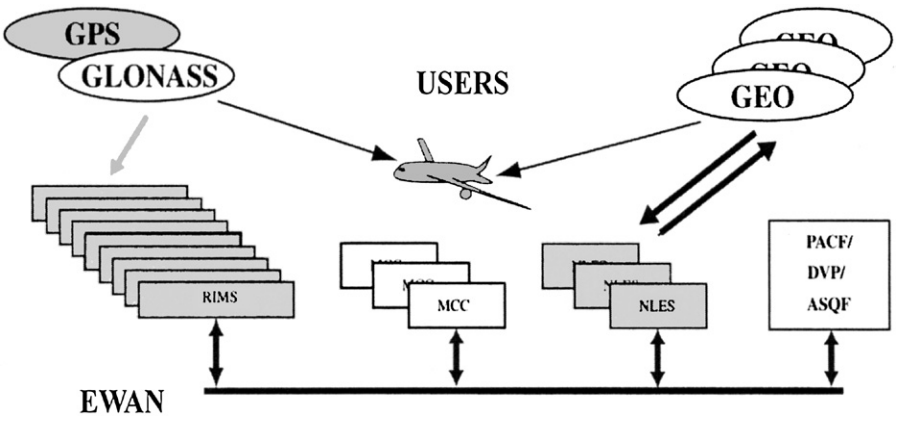


Fig. 8.2 European Global Navigation Overlay System architecture.

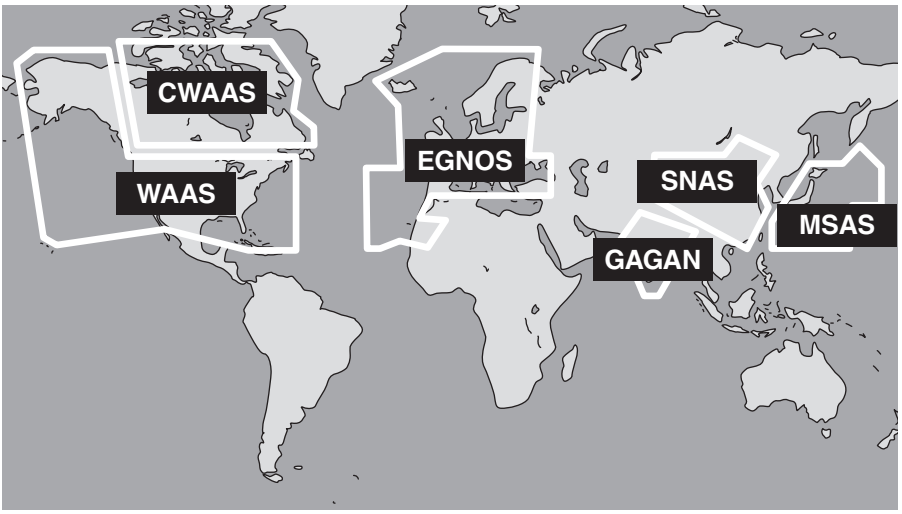


Fig. 8.3 Current and planned SBAS service areas.

TABLE 8.1. Worldwide SBAS System Coverage

Country	Acronym	Title
United States	WAAS	Wide-Area Augmentation System
Europe	EGNOS	European Geostationary Navigation Overlay System
Japan	MSAS	MTSAT Satellite-Based Augmentation System
Canada	CWAAS	Canadian Wide-Area Augmentation System
China	SNAS	Satellite Navigation Augmentation System
India	GAGAN	GPS & GEO Augmented Navigation

[DVP] is the Development and Verification Platform
[ASQF] is the Application-Specific Qualification Facility
[EWAN] is the EGNOS Wide Area (communication) Network

8.2.3.3 Other SBAS Service areas of current and future SBAS systems are mapped in Fig. 8.3, and the acronyms are listed in Table 8.1.

8.3 GEO WITH L1L5 SIGNALS

The SBAS uses GEO satellites to relay correction and integrity information to users. A secondary use of the GEO signal is to provide users with a GPS-like ranging source. The ranging signal is generated on the ground and provided via C-band uplink to the GEO, where the navigation payload translates

the uplinked signal to an L1 downlink frequency. The GEO incorporates an additional C-band downlink to provide ionospheric delay observations to the GEO uplink ground station. The GEO Communication and Control Segment (GCCS) is a second-generation satellite and uplink subsystem that adds new L1L5 GEOs and ground stations to SBAS (Fig. 8.4).

A key feature of GCCS is the addition of a second independently generated and controlled uplink signal. In contrast to SBAS, which uplinks and controls a single C-band signal, GCCS uplinks two independent C-band signals, which are translated to L1 and L5 downlink signals. Closed-loop control of the GEO's L1 and L5 broadcast signals in space is necessary to ensure that the algorithms compensate for various sources of uplink divergence between the code and carrier, including uplink ionospheric delay, uplink Doppler, and divergence due to carrier frequency translation errors induced by the GEO's transponder.

Raytheon Company has developed a subsystem for GCCS uplink signal generation under a subcontract to Lockheed Martin. GCCS added new GEOs to the SBAS, which Raytheon developed under contract with the FAA. SBAS is a GPS-based navigation system that is intended to become the primary navigational aid for aviation during all phases of flight.

The SBAS makes use of a network of WRSs distributed throughout the United States. These reference stations collect pseudorange measurements and send them to the SBAS WMSs. The master stations process the data to provide correction and integrity information for each GEO and GPS satellite in view. The corrections information includes satellite ephemeris errors, clock bias, and ionospheric estimation data. The corrections from the WMS are sent to the GUS for uplink to the GEO.

The GUS receives SBAS messages from the WMS, adds forward error correction (FEC) encoding, and transmits the messages via C-band uplink to the GEO satellite for broadcast to SBAS users. The GUS uplink signal uses the GPS standard positioning service waveform (C/A code, binary phase-shift keying [BPSK] modulation); however, the data rate is higher (250 bps). The 250 bits of data are encoded with a one-half rate convolutional code, resulting in a 500-sps transmission rate.

A key feature of GCCS is that satellite broadcasts are available at both the GPS L1 and L5 frequencies. Unlike the early designs for SBAS broadcasts, which utilize a single uplink signal frequency translated into two downlinks, GCCS uplinks two independent C-band signals, which the transponder translates in frequency and broadcasts as independent L1 and L5 downlink signals. Figure 8.4 provides a top-level view of the GCCS architecture.

For the L1 loop, each symbol is modulated by the C/A code, a 1.023×106-cps pseudorandom sequence to provide a spread-spectrum signal. This signal is then BPSK modulated by the GUS onto an intermediate frequency (IF) carrier, up-converted to a C-band frequency, and uplinked to the GEO. The satellite's navigation transponder translates the signal in frequency to an L-band (GPS L1) downlink frequency. The GUS monitors the L1 downlink

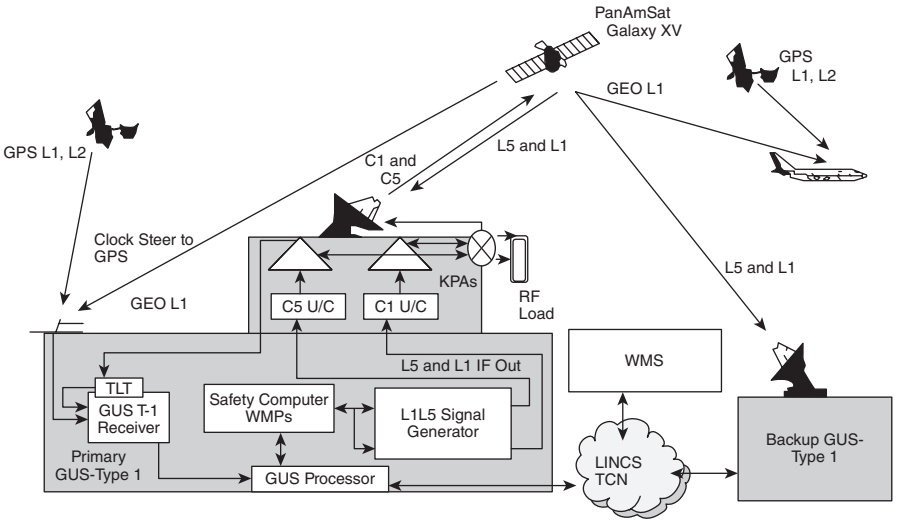


Fig. 8.4 GCCS top-level view. TLT = Test Loop Translator; KPA = Klystron Power Amplifier; LINCS = Local Information Network Communication System; TCN = Terrestrial Communication Network; RF = radio frequency.

signal from the GEO to provide closed-loop control of the code and L1 carrier. When properly controlled, the SBAS GEO provides ranging signals, as well as GPS corrections and integrity data, to end users.

The L5 spread-spectrum signal is generated by modulating each message symbol with a 10.23×106-cps pseudorandom code, which is an order of magnitude longer than that of the L1 C/A-code. As with L1, the L5 signal is then BPSK modulated onto an IF carrier, up-converted to a C-band frequency, and uplinked to the GEO. The GEO transponder will independently translate the second uplink signal to the L-band for broadcast to SBAS end users. Use of two independent broadcast signals creates unique challenges in estimating biases and maintaining coherency between the two signals.

An important aspect of the downlink signals is coherence between the code and carrier frequency. To ensure code-carrier coherency, closed-loop control algorithms, implemented in the safety computer's SBAS message processors WAAS Message Processors (WMPs), are used to maintain the code chipping rate and carrier frequency of the received L1 signal at a constant ratio of 1:1540. The second L-band (L5) downlink is used by the control algorithms to estimate and correct for ionospheric delay on the uplink signal. Control algorithms also correct for other uplink effects such as Doppler, equipment delays, and transponder offsets in order to maintain the correct Doppler and ionospheric divergence as observed by the user.

Closed-loop control of each signal is required to maintain coherence between its code and carrier frequency, as described above. With two independent signal paths, it is also required that coherence between the two carriers

be maintained for correct ionospheric delay estimation. The control-loop algorithms "precorrect" the code phase, carrier phase, and carrier frequency of the L1 and L5 signals to remove uplink effects such as ionospheric delays, uplink Doppler, equipment delays, and frequency offsets. In addition, differential biases between the L1 and L5 signals must be estimated and corrected.

Each control algorithm contains two Kalman filters and two control loops. One Kalman filter estimates the ionospheric delay and its rate of change from L1 and L5 pseudorange measurements. The second Kalman filter estimates range, range rate, range acceleration, and acceleration rate from raw pseudorange measurements. Range estimates are adjusted for ionospheric delay, as estimated by the first Kalman filter. Each code control loop generates a code chip rate command and chip acceleration command to compensate for uplink ionospheric delay and for the uplink Doppler effect. Each frequency control loop generates a carrier frequency command and a frequency rate command. A final estimator is used to calculate bias between the L1 and L5 signals.

Results of laboratory tests utilizing live L1L5 hardware elements and simulated satellite effects follow.

8.3.1 GEO Uplink Subsystem Type 1 (GUST) Control Loop Overview

The primary GUST control loop functional block diagram is shown in Fig. 8.5. The backup GUST control loop is similar to the primary GUST control loop except that the uplink signal is radiated into a dummy load. The operation of the backup GUST control loop is different from the primary GUST because of the latter.

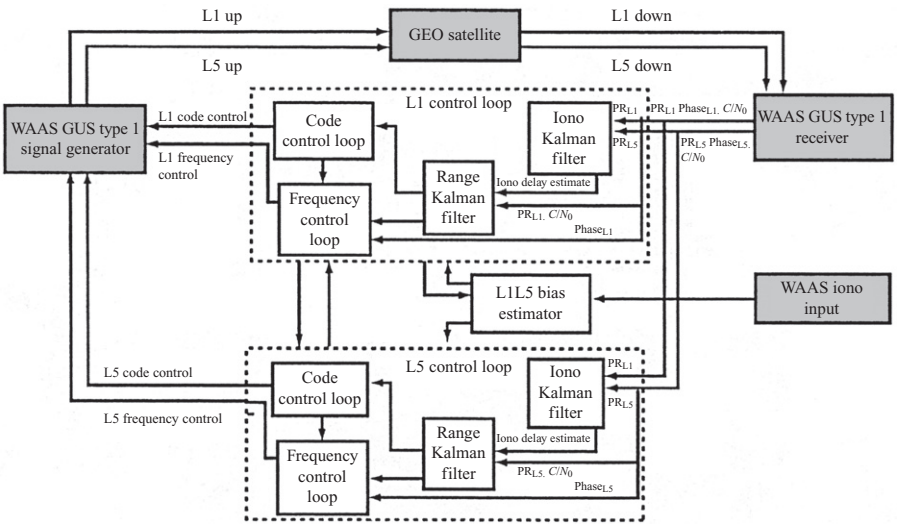


Fig. 8.5 Primary GUST control loop functional block diagram.

Each of the L1 and L5 control loops in the primary GUST consists of an iono Kalman filter, a range Kalman filter, a code control function, and a frequency control function. In addition, there is an L1L5 bias estimation function. These control loop functions reside inside the safety computer. The external inputs to the control loop algorithm are the pseudorange, carrier phase, Doppler, and carrier-to-noise ratio from the receiver.

8.3.1.1 Ionospheric Kalman Filters The L1 and L5 ionospheric (iono) Kalman filters are two-state filters:

$$\mathbf{x} = \begin{bmatrix} \text{iono delay} \\ \text{iono delay rate} \end{bmatrix}.$$

During every 1-s timeframe in the safety computer, the ionospheric Kalman filter states and the covariance are propagated. The equations for Kalman filter propagation are given in Chapter 10, Table 10.1.

The L1 filter measurement is formulated as follows:

$$z = \frac{(\rho_{\text{RL1}} - d_{\text{L1}}) - (\rho_{\text{RL5}} - d_{\text{L5}})}{(1 - \text{L1freq})^2 / (\text{L5freq})^2},$$

where ρ_{RL1} is the L1 pseudorange, ρ_{RL5} is the L5 pseudorange, d_{L1} and d_{L5} are the predetermined L1 and L5 downlink path hardware delays, L5freq is the L1 nominal frequency of 1575.42 MHz, and L5freq is the L5 nominal frequency of 1176.45 MHz.

The L5 ionospheric Kalman filter design is similar to that for L1, with the filter measurement as follows:

$$z = \frac{(\rho_{\text{RL1}} - \text{d}_{\text{L1}}) - (\rho_{\text{RL5}} - d_{\text{L5}})}{(\text{L5freq})^2 / (\text{L1freq})^2 - 1}. \tag{8.1}$$

8.3.1.2 Range Kalman Filter The L1 and L5 range Kalman filters use four state variables:

$$\overset{\text{def}}{\mathbf{x}} = \begin{bmatrix} \text{range} \\ \text{range rate} \\ \text{acceleration} \\ \text{acceleration rate} \end{bmatrix}. \tag{8.2}$$

During every 1-s timeframe in the safety computer, the range Kalman filter states and their covariance of uncertainty are propagated (predicted) in the Kalman filter.

After the filter propagation, if L1 pseudorange is valid, the L1 range estimate and covariance are updated in the Kalman filter using the L1 pseudorange measurement correction. Likewise, if L5 pseudorange is valid, the L5

range estimate and covariance are updated in the Kalman filter using the L5 pseudorange measurement correction.

The L1 range Kalman filter measurement is

$$z = \rho_{\mathrm{RL1}} - d_{\mathrm{L1}} - I_{\mathrm{L1}},$$

where ρ_{RL1} is the L1 pseudorange, $d_{\mathrm{L1}}$is the predetermined L1 downlink path hardware delays, and I_{L1} is the L1 ionospheric delay estimate.

Likewise, the L5 range Kalman filter measurement is

$$z = \rho_{\mathrm{RL1}} - d_{\mathrm{L1}} - I_{\mathrm{L1}},$$

where ρ_{RL5} is the L5 pseudorange, d_{L5} is the predetermined L5 downlink path hardware delays, and I_{L5} is the L5 ionospheric delay estimate. The required Kalman filter equations are given in Table 10.1.

8.3.1.3 Code Control Function The L1 and L5 code control functions compute the corresponding code chip rate commands and the chip acceleration commands to be sent to the signal generator. The signal generator adjusts its L1 and L5 chip rates according to these commands. The purpose of code control is to compensate for any initial GEO range estimation error, the iono delay on the uplink C-band signal, and the Doppler effects due to the GEO movement on the uplink signal code chip rate. This compensation will ensure that the GEO signal code phase deviation is within the required limit.

The receiver and signal generator timing 1-pps (pulse per second) errors also affect the GEO signal code phase deviation. These errors are compensated separately by the clock steering algorithm [12].

Measurement errors in the predetermined hardware delays of the two signal paths (both uplink and downlink) will result in additional code phase deviation for the GEO signal due to the closed-loop control. This additional code phase deviation will be interpreted as GEO satellite clock error by the master station's GEO orbit determination (OD). Since the clock steering algorithm will use the SBAS broadcast type 9 message GEO clock offset as part of the input to the clock steering controller [12], the additional code phase deviation due to common measurement errors will be compensated for by the clock steering function.

There are several inputs to the code control function: the uplink range, the projected range of the GEO for the next 1-s timeframe, the estimated iono delay, and so on. The uplink range is the integration of the commanded chip rate, and this integration is performed in the safety computer. The commanded chip acceleration is computed on the basis of the estimated acceleration from the Kalman filter (see Table 10.1).

8.3.1.4 Frequency Control Function The L1 and L5 frequency control functions compute the corresponding carrier frequency commands and the

frequency change rate (acceleration) commands to be sent to the signal generator. The signal generator adjusts the L1 and L5 IF outputs according to these commands. The purpose of frequency control is to compensate for the Doppler effects due to the GEO movement on the carrier of the uplink signal, the effect of iono rate on the uplink carrier, and the frequency offset of the GEO transponders. This function also continuously estimates the GEO transponder offset, which could drift during the lifetime of the GEO satellite.

8.3.1.5 L1L5 Bias Estimation Function This function estimates the bias between the L1 and L5 that is due to differential measurement errors in the predetermined hardware delays of the two signal paths. If not estimated and compensated, the bias between L1 and L5 will be indistinguishable from iono delay, as shown in the equations below. L1 and L5 pseudorange can be expressed as

$$\rho_{\mathrm{RL1}} = \rho + I_{\mathrm{L1}} + \text{true } d_{\mathrm{L1}} + \text{clock error} + \text{tropo delay}, \tag{8.3}$$

$$\rho_{\mathrm{RL5}} = \rho + I_{\mathrm{L5}} + \text{true } d_{\mathrm{L5}} + \text{clock error} + \text{tropo delay}, \tag{8.4}$$

where R is the true range, I_{L1} is the true L1 iono delay, I_{L5} is the true L5 iono delay, true d_{L1} is the true L1 downlink path hardware delay, and true true d_{L5} is the true L5 downlink path hardware delay.

This becomes

$$\rho_{\mathrm{RL1}} - d_{\mathrm{L1}} = \rho + I_{\mathrm{L1}} + \text{true } d_{\mathrm{L1}} + \text{clock error} + \text{tropo delay} - d_{\mathrm{L1}}, \tag{8.5}$$

$$\rho_{\mathrm{RL5}} - d_{\mathrm{L5}} = \rho + I_{\mathrm{L5}} + \text{true } d_{\mathrm{L5}} + \text{clock error} + \text{tropo delay} - d_{\mathrm{L5}}, \tag{8.6}$$

where d_{L1} is the predetermined (measured) L1 downlink path hardware delay and d_{L5} is the predetermined (measured) L5 downlink path hardware delay.

Let $\Delta d_{\mathrm{L1}} = \text{true } d_{\mathrm{L1}} - d_{\mathrm{L1}}$ and $\Delta d_{\mathrm{L5}} = \text{true } d_{\mathrm{L5}} - d_{\mathrm{L5}}$. The measurement for the L1 iono Kalman filter becomes

$$z = \frac{(\rho_{\mathrm{RL1}} - d_{\mathrm{L1}}) - (\rho_{\mathrm{RL5}} - d_{\mathrm{L5}})}{(1 - \mathrm{L1freq})^2 / (\mathrm{L5freq})^2}, \tag{8.7}$$

$$= I_{\mathrm{L1}} + \frac{(\Delta d_{\mathrm{L1}} - \Delta d_{\mathrm{L5}})}{(1 - \mathrm{L1freq})^2 / (\mathrm{L5freq})^2}. \tag{8.8}$$

The term $(\Delta d_{\mathrm{L1}} - \Delta d_{\mathrm{L5}})/(1 - \mathrm{L1freq})^2/(\mathrm{L5freq})^2$ is the differential L1L5 bias term, and it becomes an error in the L1 iono delay estimation. The L5 iono Kalman filter is similarly affected by the L1L5 bias term.

8.3.1.6 L1L5 Bias Estimation Function The GEO's broadcast code-carrier coherence (CCC) requirement is specified in the WAAS System Specification and in Appendix A of Ref. 13. The WAAS System Specification states the requirement as:

The lack of coherence between the broadcast SIS L1 carrier phase and its respective code phase shall be limited in accordance with the following equation such that the standard deviation over T seconds of the error due to 100-second carrier-smoothing of the code-based pseudorange is less than one carrier wavelength. This equation does not include code-carrier divergence due to ionospheric refraction in the downlink propagation path.

$$\sigma_{(T\,\mathrm{sec})}\left[\rho_{\mathrm{RL1}}(t)-\overline{\rho_{\mathrm{RL1}}(t)}\right]>0.19\ m.$$

The term ρ_{RL1} is the L1 pseudorange that would be measured by a noiseless receiver, and $\overline{\rho_{\mathrm{RL1}}(t)}$ is the carrier smoothed L1 pseudorange. WAAS CCC is defined over an interval of T – 86,400 s (1 day). Carrier smoothing is performed over a 100-s interval. Note that a noiseless receiver, as used here, means the values that would be measured by a hypothetical noiseless receiver in parallel with the ground station receiver. Alternately, because code-carrier divergence due to ionospheric refraction in the downlink signal path is excluded, the noiseless receiver may be considered to be at the focus of the GEO transmit antenna.

The WAAS CCC equation above is interpreted from the Radio Technical Commission for Aeronautics (RTCA) Minimum Operational Performance Standards (MOPS) equation [13] and is based on the need to limit errors associated with carrier smoothing of the code-based pseudorange. Code-carrier coherence results for two WAAS GEOs are provided in Table 8.2. The results indicate that the control loop algorithm performance meets WAAS requirements [3].

8.3.1.7 Carrier Frequency Stability Carrier frequency stability is a function of the uplink frequency standard, GUS signal generator, and GEO satellite transponder. The GEO's short-term carrier frequency stability requirement is specified in the WAAS System Specification and Appendix A of Ref. 13. It states: "The short term stability of the carrier frequency (square root of the Allan variance) at the input of the user's receiver antenna shall be better than 5×10^{-11} over 1 to 10 s, excluding the effects of the ionosphere and Doppler."

The Allan variance [14] is calculated on the double difference of L1 phase data divided by the center frequency over 1 – 10 s. Effects of smoothed ionosphere and Doppler are compensated for in the data prior to this calculation.

TABLE 8.2. Code-Carrier Coherence Results

		Code Carrier Coherence Requirement
GEO Satellite	Date	<1 cycle
CRW	August 25, 2009	0.65
CRE	August 25, 2009	0.99

8.4 GUS CLOCK STEERING ALGORITHM

Presently, the SBAS WMS calculates SBAS network time (WNT) and estimates clock parameters (offset and drift) for each satellite. The GEO uplink system (GUS) clock is an independent free running clock. However, the GUS clock must track WNT (GPS time) to enable accurate ranging from the GEO SIS. Therefore, a clock steering algorithm is necessary. The GUS clock steering algorithms reside in the SBAS message processor (WMP). The SBAS type 9 message (GEO navigation message) is used as input to the GUS WMP, provided by the WMS.

The GUS clock is steered to the GPS time epoch (see also Fig. 8.6). The GUS receiver clock error is the deviation of its 1-s pulse from the GPS epoch. The clock error is computed in the GUS processor by calculating the user position error by combining (in the least-squares sense, weighted with expected error statistics) multiple satellite data (pseudorange residuals called MOPS *residuals*) [13] into a position error estimate with respect to surveyed GUS position. The clock steering algorithm is initialized with the SBAS type 9 message (GEO navigation message). This design keeps the GUS receiver clock 1 pps synchronized with the GPS time epoch. Since the 10-MHz frequency standard is the frequency reference for the receiver, its frequency

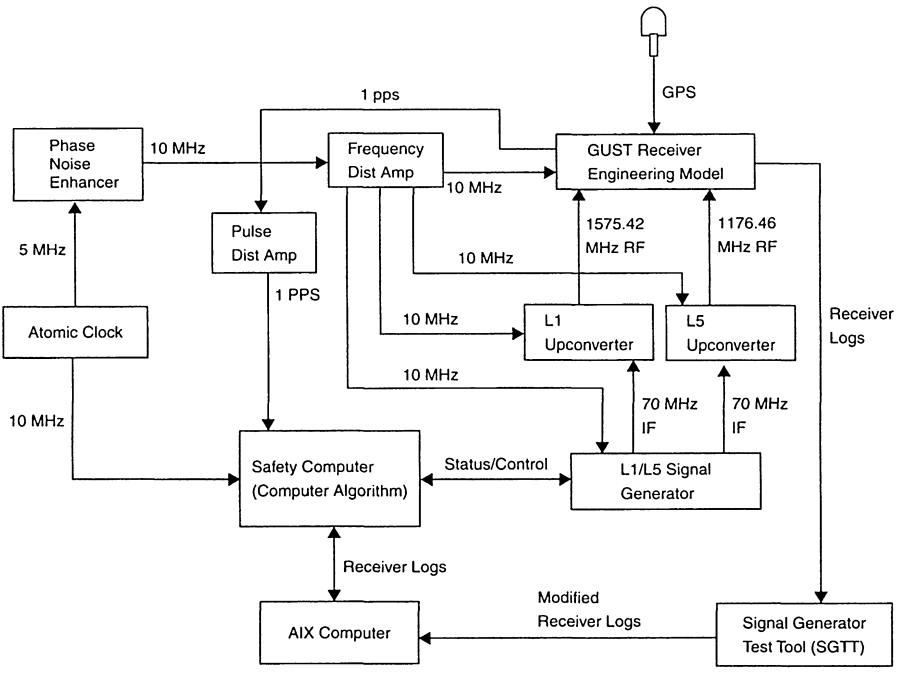


Fig. 8.6 Control loop test setup.

output needs to be controlled so that the 1 pps is adjusted. A proportional, integral, and differential (PID) controller has been designed to synchronize to the GPS time at GUS locations.

This algorithm also decouples the GUS clock from orbit errors and increases the observability of orbit errors in the OD filter in the correction processor of the WMS. It also synchronizes GUS clocks at all GUS locations to GPS time.

In the initial 24 h after a GUS becomes primary, the clock steering algorithm uses SBAS type 9 messages from the WMS to align the GEO's epoch with the GPS epoch. The SBAS type 9 message contains a term referred to as $a_{G/0}$ or clock offset. This offset represents a correction, or time difference, between the GEO's epoch and SBAS network time (WNT). WNT is the internal time reference scale of SBAS and is required to track the GPS timescale, while at the same time providing users with the translation to UTC. Since GPS master time is not directly obtainable, the SBAS architecture requires that WNT be computed at multiple WMSs using potentially differing sets of measurements from potentially differing sets of receivers and clocks (SBAS reference stations). WNT is required to agree with GPS to within 50 ns. At the same time, the WNT to UTC offset must be provided to the user, with the offset being accurate to 20 ns. The GUS calculates local clock adjustments. On the basis of these clock adjustments, the frequency standard can be made to speed up or slow the GUS clock. This will keep the total GEO clock offset within the range allowed by the SBAS type 9 message so that users can make the proper clock corrections in their algorithms [15, 16].

After the initial 24 h, once the GUS clock is synchronized with WNT, a second steering method of clock steering is used. The algorithm now uses the composite of the MOPS [13] solution for the receiver clock error, and the average of the $a_{G/0}$, and the average of the MOPS solution as the input to the clock steering controller.

8.4.1 Receiver Clock Error Determination

Determination of receiver clock error is based on the user position solution algorithm described in the SBAS MOPS. The clock bias (C_b) is a resultant of the MOPS weighted least-squares solution.

Components of the weighted least-squares solution are the observation matrix (**H**), the measurement weighting matrix (**W**), and the MOPS residual column vector ($\Delta\boldsymbol{\rho}$) The weighted gain matrix (**K**) is calculated using **H** and **W** (see Eq. 2.34):

$$\mathbf{K} = \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}. \tag{8.9}$$

From this, the column vector for the user position error and the clock bias solution is

$$\Delta\mathbf{X} = \mathbf{K}\Delta\boldsymbol{\rho} \tag{8.10}$$

$$\Delta\mathbf{X} = \left(\mathbf{H}^T\mathbf{W}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}\Delta\boldsymbol{\rho}, \tag{8.11}$$

where

$$\Delta\mathbf{X} = \begin{bmatrix} \Delta X(U) \\ \Delta X(E) \\ \Delta X(N) \\ C_{\mathrm{b}} \end{bmatrix} \tag{8.12}$$

and $\Delta X(U)$ is the up error, $\Delta X(E)$ is the east error, $\Delta X(N)$ is the north error, and C_{b} is the clock bias or receiver clock error.

The $n \times 4$ observation matrix ($\mathbf{H}$) is computed in up–east–north (UEN) reference frame using the LOS azimuth (Az_i) and LOS elevation (El_i) from the GUS omni antenna to the space vehicle (SV). The value n is the number of satellites in view. The formula for calculating the observation matrix is

$$\mathbf{H} = \begin{bmatrix} \cos(\mathrm{El}_1)\cos(\mathrm{Az}_1) & \cos(\mathrm{El}_1)\sin(\mathrm{Az}_1) & \sin(\mathrm{El}_1) & 1 \\ \cos(\mathrm{El}_2)\cos(\mathrm{Az}_2) & \cos(\mathrm{El}_2)\sin(\mathrm{Az}_2) & \sin(\mathrm{El}_2) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \cos(\mathrm{El}_n)\cos(\mathrm{Az}_n) & \cos(\mathrm{El}_n)\sin(\mathrm{Az}_{\mathrm{n}}) & \sin(\mathrm{El}_n) & 1 \end{bmatrix}. \tag{8.13}$$

The $n \times n$ weighting matrix ($\mathbf{W}$) is a function of the total variance $\left(\sigma_i^2\right)$ of the individual satellites in view. The inverse of the weighting matrix is

$$\mathbf{W}^{-1} = \begin{bmatrix} \sigma_1^2 & 0 & 0 & \vdots & 0 \\ 0 & \sigma_2^2 & 0 & \vdots & 0 \\ 0 & 0 & \sigma_3^2 & \vdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sigma_n^2 \end{bmatrix}. \tag{8.14}$$

The equation to calculate the total variance $\left(\sigma_i^2\right)$ is

$$\sigma_i^2 = \left(\frac{\mathrm{UDRE}_i}{3.29}\right)^2 + \left(\frac{F_{\mathrm{pp}i} \times \mathrm{GIVE}_i}{3.29}\right)^2 + \sigma_{\mathrm{L1},nmp,i}^2 + \frac{\sigma_{\mathrm{tropo},i}^2}{\sin^2 \mathrm{El}_i}. \tag{8.15}$$

The algorithms for calculating user differential range error (UDRE_i), user grid ionospheric vertical error (GIVE_i), LOS obliquity factor ($F_{\mathrm{pp}i}$), standard deviation of uncertainty for the vertical troposphere delay model (a tropo_i), and the standard deviation of noise and multipath on the L1 omni pseudorange $\sigma_{\mathrm{L1}nmp,i}$ are found in the SBAS MOPS [13].

The MOPS residuals ($\Delta\rho$) are the difference between the smoothed MOPS measured pseudorange ($PR_{\mathrm{M},i}$) and the expected pseudorange ($PR_{\mathrm{corr},i}$)

$$\Delta\rho = \begin{bmatrix} PR_{\mathrm{M},1} - PR_{\mathrm{corr},1} \\ PR_{\mathrm{M},2} - PR_{\mathrm{corr},2} \\ PR_{\mathrm{M},3} - PR_{\mathrm{corr},3} \\ \vdots \end{bmatrix}. \tag{8.16}$$

The MOPS measured pseudorange ($PR_{\mathrm{M},i}$) in earth-centered, earth-fixed (ECEF) reference is corrected for earth rotation, for SBAS clock corrections, for ionospheric effects, and for tropospheric effects. The equation to calculate ($PR_{\mathrm{M},i}$) is

$$PR_{\mathrm{M},i} = PR_{\mathrm{L},i} + \Delta PR_{\mathrm{CC},i} + \Delta PR_{\mathrm{FC},i} + \Delta PR_{\mathrm{ER},i} - \Delta PR_{\mathrm{T},i} - \Delta PR_{\mathrm{I},i}. \tag{8.17}$$

The algorithms used to calculate smoothed L1 omni pseudorange ($PR_{\mathrm{L},i}$), pseudorange clock correction ($\Delta PR_{\mathrm{CC},i}$) pseudorange fast correction ($\Delta PR_{\mathrm{FC},i}$), pseudorange earth rotation correction ($\Delta PR_{\mathrm{ER},i}$) pseudorange troposphere correction ($\Delta PR_{\mathrm{T},i}$) and pseudorange ionosphere correction ($\Delta PR_{\mathrm{I},i}$) are found in the SBAS MOPS [13].

Expected pseudorange ($PR_{\mathrm{corr},i}$) ECEF, at the time of GPS transmission is computed from broadcast ephemeris corrected for fast and long-term corrections. The calculation is

$$PR_{\mathrm{corr},i} = \sqrt{(X_{\mathrm{corr},i} - X_{\mathrm{GUS}})^2 + (Y_{\mathrm{corr},i} - Y_{\mathrm{GUS}})^2 + (Z_{\mathrm{corr},i} - Z_{\mathrm{GUS}})^2}. \tag{8.18}$$

The fixed-position parameters of the WRE (X_{GUS}, Y_{GUS}, Z_{GUS}) are site specific.

8.4.2 Clock Steering Control Law

In the primary GUS, the clock steering algorithm is initialized with SBAS type 9 message (GEO navigation message). After the initialization, composite of MOPS solution and type 9 message for the receiver clock error is used as the input to the control law (see Fig. 8.7). For the backup GUS, the MOPS solution for the receiver clock error is used as the input to the control law (see Fig. 8.8).

For both the primary and backup clock steering algorithm, the control law is a PID controller. The output of the control law will be the frequency adjustment command. This command is sent to the frequency standard to adjust the atomic clock frequency. The output frequency to the receiver causes the 1 pps to approach the GPS epoch. Thus, a closed-loop control of the frequency standard is established.

8.5 GEO ORBIT DETERMINATION (OD)

The purpose of WAAS is to provide pseudorange and ionospheric corrections for GPS satellites to improve the accuracy for the GPS navigation user and to protect the user with "integrity." Integrity is the ability to provide timely warn-

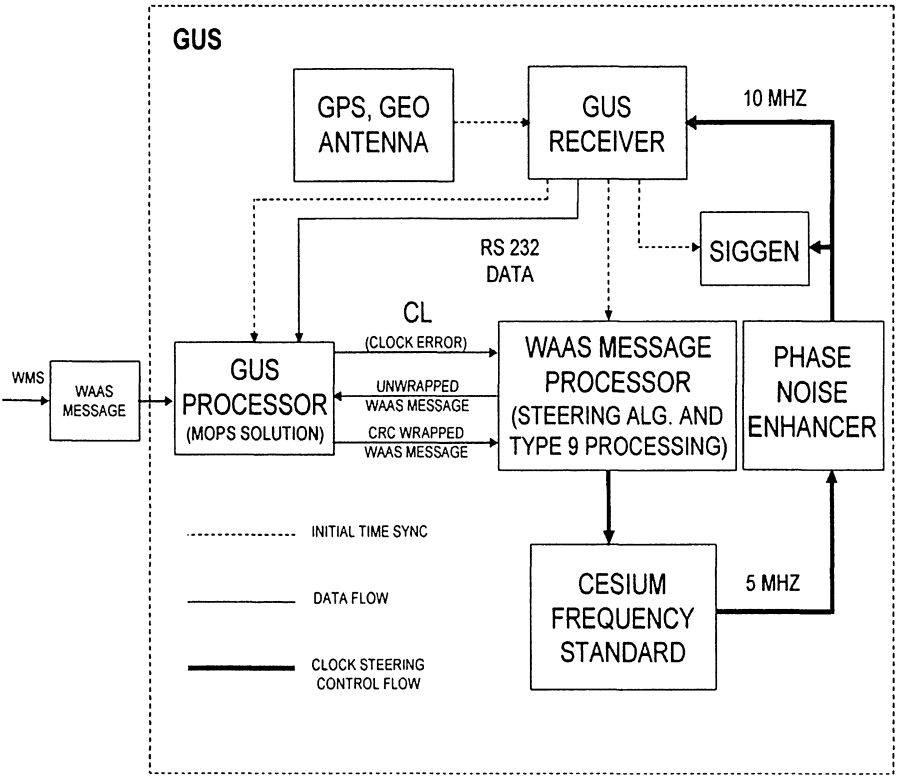


Fig. 8.7 Primary GUS clock steering.

ings to the user whenever any navigation parameters estimated using the system are outside tolerance limits. WAAS may also augment the GPS constellation by providing additional ranging sources using GEO satellites that are being used to broadcast the WAAS signal.

The two parameters having the most influence on the integrity bounds for the broadcast data are UDRE for the pseudorange corrections and GIVE for the ionospheric corrections. With these, the onboard navigation system estimates the horizontal protection limit (HPL) and the vertical protection limit (VPL), which are then compared to the horizontal alert limit (HAL) and the vertical alert limit (VAL) requirements for the particular phase of flight involved, that is, oceanic/remote, en route, terminal, NPA, and PA. If the estimated protection limits are greater than the alert limits, the navigation system is declared unavailable. Therefore, the UDRE and GIVE values obtained by the WAAS (in concert with the GPS and GEO constellation geometry and reliability) essentially determine the degree of availability of the WADGPS navigation service to the user.

The WAAS algorithms calculate the broadcast corrections and the corresponding UDREs and GIVEs by processing the satellite signals received by the network of ground stations. Therefore, the expected values for UDREs

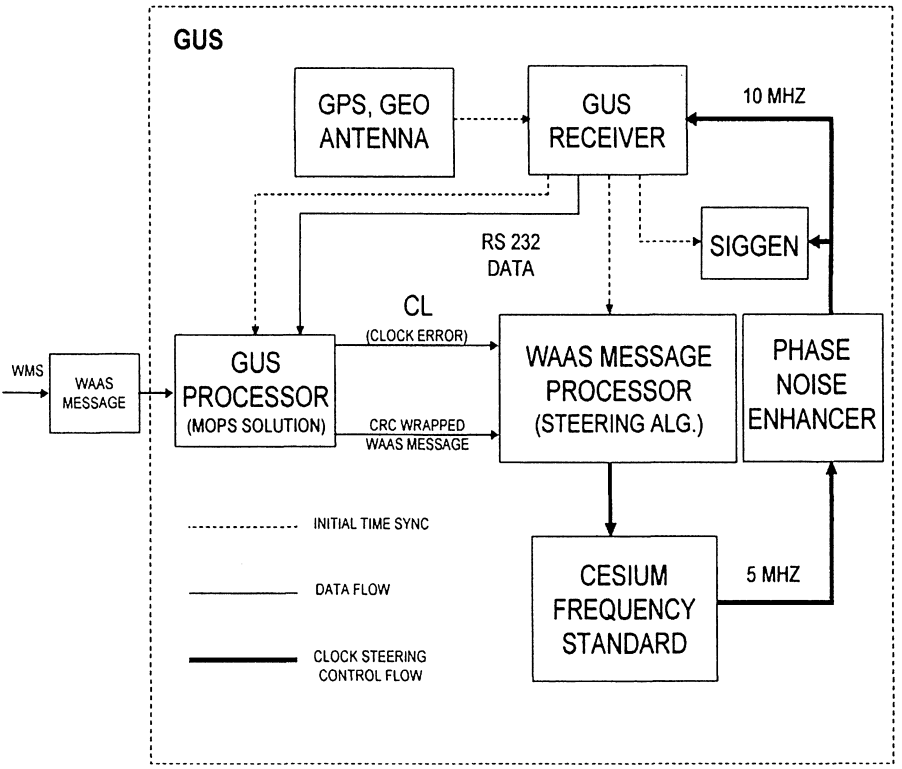


Fig. 8.8 Backup GUS clock steering.

and GIVEs are dependent on satellite and station geometries, satellite signal and clock performance, receiver performance, environmental conditions (such as multipath and ionospheric storms), and algorithm design [17, 18].

8.5.1 OD Covariance Analysis

A full WAAS algorithm contains three Kalman filters—an OD filter, an ionospheric corrections filter, and a fast corrections filter. The fast corrections filter is a Kalman filter that estimates the GEO, GPS, and ground station clock states every second. In this section, we derive an estimated lower bound of the GEO UDRE for a WAAS algorithm that contains only the OD Kalman filter, called the UDRE(OD), where OD refers to orbit determination.

A method is proposed to approximate the UDRE obtained for a WAAS including both the OD filter and the fast corrections filter from UDRE(OD). From case studies of the geometries studied in the previous section, we obtain the essential dependence of UDRE on ground station geometry.

A covariance analysis on the OD is performed using a simplified version of the OD algorithms. The performance of the ionospheric corrections filter is

treated as perfect, and therefore, the ionospheric filter model is ignored. The station clocks are treated as if perfectly synchronized using the GPS satellite measurements. Therefore, the station clock states are ignored. This allows the decoupling of the ODs for all the satellites from each other, simplifying the OD problem to that for one satellite with its corresponding ground station geometry and synchronized station clocks. Both of these assumptions are liberal; therefore, the UDRE(OD) obtained here is a lower bound for the actual UDRE(OD). Finally, we consider only users within the service volume covered by the stations and, therefore, ignore any degradation factors depending on user location.

To simulate the Kalman filter for the covariance matrix P, the following four matrices are necessary (Table 10.1):

Φ = state transition matrix,

$\mathbf{H}$ = measurement sensitivity matrix,

$\mathbf{Q}$ = process noise covariance matrix and

$\mathbf{R}$ = measurement noise covariance matrix

The methods used to determine these matrices are described below.

The state vector for the satellite is

$$\mathbf{x} = \begin{bmatrix} \mathbf{r} \\ \dot{\mathbf{r}} \\ C_b \end{bmatrix},$$

where

$$\mathbf{r} \equiv [x\ y\ z]^T$$

is the satellite position in the earth-centered inertial (ECI) frame;

$$\dot{\mathbf{r}} \equiv [\dot{x},\ \dot{y}, \dot{z}]^T$$

is the satellite velocity in the ECI frame; and C_b is the satellite clock offset relative to the synchronized station clocks. Newton's second and third (gravitational) laws provide the equations of motion for the satellite:

$$\ddot{\mathbf{r}} \equiv \frac{d^2 r}{dt^2} = -\frac{\mu_E \mathbf{r}}{|\mathbf{r}|^3} + M,$$

where $\ddot{\mathbf{r}}$ is the acceleration in the EC1 frame, μ_E is the gravitational constant for the earth, and M is the total perturbation vector in the ECI frame containing all the perturbing accelerations. For this analysis, only the perturbation due to the oblateness of the earth is included. The effect of this perturbation on the behavior of the covariance is negligible, and therefore higher-order perturbations are ignored. (Note that although the theoretical model is simplified,

the process noise covariance matrix Q is chosen to be consistent with a far more sophisticated orbital model.)

Therefore,

$$M = -\frac{3}{2} J_2 \frac{\mu_E}{|\mathbf{r}|^3} \frac{a_E^2}{|\mathbf{r}|^2} \left[\mathbf{I}_{3\times 3} + 2\hat{\mathbf{z}}\hat{\mathbf{z}}^T \right] \mathbf{r},$$

where a_E is the semimajor axis of the earth-shape model, J_2 is the second zonal harmonic coefficient of the earth-shape model, and $\hat{\mathbf{z}} \equiv [0, 0, 1]^T$ [19].

The second-order differential equation of motion can be rewritten as a pair of first-order differential equations:

$$\dot{\mathbf{r}}_1 = \mathbf{r}_2, \dot{\mathbf{r}}_2 = \frac{\mu_E r_1}{|\mathbf{r}|^3} + M, \tag{8.19}$$

where $\mathbf{r}_1$ and $\mathbf{r}_2$ are vectors, which therefore gives a system of six first-order equations.

The variational equations are differential equations describing the rates of change of the satellite position and velocity vectors as functions of variations in the components of the estimation state vector. These lead to the state transition matrix Φ used in the Kalman filter. The variational equations are

$$\ddot{Y}(t) = A(t)Y(t) + B(t)\dot{Y}(t), \tag{8.20}$$

where

$$Y(t_k)_{3\times 6} \equiv \left[\left(\frac{\partial \mathbf{r}(t_k)}{\partial \mathbf{r}(t_{k-1})} \right)_{3\times 3} \left(\frac{\partial \mathbf{r}(t_k)}{\partial \dot{\mathbf{r}}(t_{k-1})} \right)_{3\times 3} \right], \tag{8.21}$$

$$\dot{Y}(t_k)_{3\times 6} \equiv \left[\left(\frac{\partial \dot{\mathbf{r}}(t_k)}{\partial \mathbf{r}(t_{k-1})} \right)_{3\times 3} \left(\frac{\partial \mathbf{r}(t_k)}{\partial \dot{\mathbf{r}}(t_{k-1})} \right)_{3\times 3} \right], \tag{8.22}$$

$$\begin{aligned} A(t)_{3\times 3} &\equiv \frac{\partial \ddot{\mathbf{r}}}{\partial \mathbf{r}} \\ &= \frac{-\mu_E}{|\mathbf{r}|^3} \left[\mathbf{I}_{3\times 3} - 3\hat{\mathbf{r}}\hat{\mathbf{r}}^T \right] - \frac{3}{2} J_2 \frac{\mu_E}{|\mathbf{r}|^3} \frac{a_E^2}{|\mathbf{r}|^2}, \\ &\quad \times \left[\mathbf{I}_{3\times 3} + 2\hat{\mathbf{z}}\hat{\mathbf{z}}^T - 10\left(\hat{\mathbf{r}}^T\hat{\mathbf{z}}^T\right)\left(\hat{\mathbf{z}}\hat{\mathbf{r}}^T + \hat{\mathbf{r}}\hat{\mathbf{z}}^T\right) + \left(10\left(\hat{\mathbf{r}}^T\hat{\mathbf{z}}\right)^2 - 5\right)\left(\hat{\mathbf{r}}\hat{\mathbf{r}}^T\right) \right], \end{aligned} \tag{8.23}$$

$$B(t)_{3\times 3} \equiv \frac{\partial \ddot{\mathbf{r}}}{\partial \dot{\mathbf{r}}} = \mathbf{0}_{3\times 3}, \tag{8.24}$$

where $\hat{\mathbf{r}} = \mathbf{r}/|\mathbf{r}|$.

Equations 8.21–8.24 are substituted into Eqs. 8.20 and 8.19, and the differential equations are solved using the fourth-order Runge–Kutta method. The time step used is a 5-min interval. The initial conditions for the GEO are specified for the particular case given and propagated forward for each time step, whereas the initial conditions for the Y terms are

$$Y(t_{k-1})_{3\times 6} = [\mathbf{I}_{3\times 3} \quad \mathbf{0}_{3\times 3}], \quad \dot{Y}(t_k)_{3\times 6} = [\mathbf{0}_{3\times 3} \quad \mathbf{I}_{3\times 3}]$$

and are reset for each time step. This is due to the divergence of the solution of the differential equation used in this method to calculate the state transition matrix for the Kepler problem.

This gives the state $\mathbf{x}^T = [\mathbf{r}_1^T \quad \mathbf{r}_2^T]$ and the state transition matrix

$$\mathbf{\Phi}_{k,k-1_{7\times 7}} = \begin{bmatrix} Y(t_k)_{3\times 6} & \mathbf{0}_{3\times 1} \\ \dot{Y}(t_k)_{3\times 6} & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{1\times 6} & \mathbf{I}_{1\times 1} \end{bmatrix} \tag{8.25}$$

for the Kalman filter.

The measurement sensitivity matrix is given by

$$H_{N\times 7} \equiv \left[\left(\frac{\partial \rho}{\partial \mathbf{r}} \right)_{N\times 3} \left(\frac{\partial \rho}{\partial \dot{\mathbf{r}}} \right)_{N\times 3} \left(\frac{\partial \rho}{\partial (ct)} \right)_{N\times 1} \right],$$

where ρ is the pseudorange for a station and N is the number of stations in view of the satellite. Note that this is essentially the same H as in the previous section. Ignoring relativistic corrections and denoting the station position by the vector $r_s \equiv [x_s \; y_s \; z_s]^{\mathrm{T}}$, the matrices above are given by

$$\frac{\partial \rho}{\partial \mathbf{r}} = \frac{[\mathbf{r} - r_s]^{\mathrm{T}}}{\mathbf{r} - r_s} \frac{\partial \mathbf{r}(t_k)}{\partial \mathbf{r}(t_{k-1})},$$

$$\frac{\partial \rho}{\partial \dot{\mathbf{r}}} = \frac{[\mathbf{r} - r_s]^{\mathrm{T}}}{\mathbf{r} - r_s} \frac{\partial \mathbf{r}(t_k)}{\partial \dot{\mathbf{r}}(t_{k-1})},$$

and

$$\frac{\partial \rho}{\partial (ct)} = 1. \tag{8.26}$$

The station position is calculated with the WGS84 model for the earth and converted to the ECI frame using the J2000 epoch (see Appendix B).

These are then combined with the measurement noise covariance matrix R and the process noise covariance matrix Q to obtain the Kalman filter equations for the covariance matrix P, as shown in Table 10.1.

The initial condition, $P_0(+)$, and Q are chosen to be consistent with the WAAS algorithms. The value of R is chosen by matching the output of the GEO covariance for AOR-W with $R = \sigma^2\mathbf{I}$ and is used as the input R for all other satellites and station geometries (note that this therefore gives approximate results). This corresponds to carrier phase ranging for the stations. The results corresponding to the value of R for code ranging are also presented.

From this covariance, the lower bound on the UDRE is obtained by

$$\text{UDRE} \geq \text{EMRBE} + K_{ss}\sqrt{\text{tr}(P)},$$

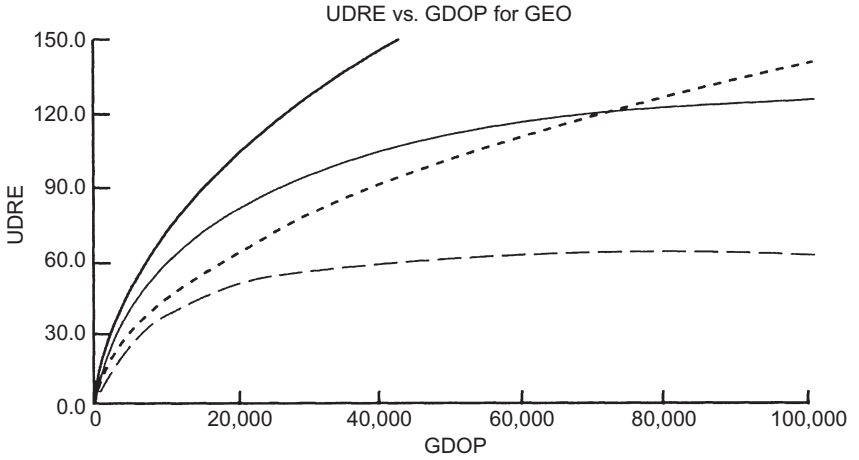


Fig. 8.9 Relationship between UDRE and GDOP.

where EMRBE is the estimated maximum range and bias error. EMRBE = 0, $K_{ss} = 3.29$ will bring the 0.999 level of bounding for the UDRE. Finally, since the message is broadcast every second, $\Delta t = 1$, so the trace can be used for the velocity components as well.

Figure 8.9 shows the relationship between UDRE and geometric dilution of precision (GDOP) for various GEO satellites and WRS locations. Table 8.3 describes the various cases considered in this analysis.

The numerical values used for the filter are as follows (all units are Système International [SI]):

Earth parameters:

$$\mu_E = 3.98600441 \times 10^{14} \quad J_2 = 1082.63 \times 10^{-6},$$
$$a_E = 6,378,137.0 \quad b_E = 6,356,752.3142.$$

8.6 GROUND-BASED AUGMENTATION SYSTEM (GBAS)

8.6.1 Local-Area Augmentation System (LAAS)

The LAAS (near airports) is being designed to provide DGPS corrections in support of navigation and landing systems. The system provides monitoring functions via LAAS Ground Facility (LGF) and includes individual measurements, ranging sources, reference receivers, navigation data, data broadcast, environment sensors, and equipment failures. Each identified monitor has a corresponding system response including alarms, alerts, and service alerts (see Fig. 8.10.)

TABLE 8.3. Cases Used in Geometry-per-Station Analysis

Case	UDRE	GDOP	Satellite	Geometry
1	17.9	905	AOR-W	WAAS stations (25), 21 in view
2	45.8	2,516	AOR-W	4 WAAS stations (CONUS)"
3	135.0	56,536	AOR-W	4 WAAS stations (NE)"
4	4.5	254	AOR-W	WAAS stations + Santiago
5	5.8	212	AOR-W	WAAS stations + London
6	4.0	154	AOR-W	WAAS stations -I- Santiago + London
7	7.5	439	AOR-W	4 WAAS stations (CONUS) + Santiago
8	8.6	337	AOR-W	4 WAAS stations (CONUS) + London
9	6.6	271	AOR-W	4 WAAS stations (CONUS) + Santiago + London
10	47.7	2,799	AOR-W	4 WAAS stations (NE) + Santiago
11	21.5	1,405	AOR-W	4 WAAS stations (NE) + London
12	16.4	1,334	AOR-W	4 WAAS stations (NE) + Santiago + London
13	28.5	1,686	POR	WAAS stations (25), 8 in view
14	45.4	3,196	POR	WAAS stations, Hawaii
15	31.1	1,898	POR	WAAS stations, Cold Bay
16	55.0	4,204	POR	WAAS stations, Hawaii, Cold Bay
17	6.7	257	POR	WAAS stations + Sydney
18	8.3	338	POR	WAAS stations + Tokyo
19	6.7	257	POR	WAAS stations + Sydney + Tokyo
20	21.0	1,124	MTSAT	MSAS stations, 8 in view
21	22.0	1,191	MTSAT	MSAS stations—Hawaii
22	24.9	1,407	MTSAT	MSAS stations—Australia
23	54.6	4,149	MTSAT	MSAS stations—Hawaii, Australia
24	22.0	1,198	MTSAT	MSAS stations—Ibaraki
25	29.0	1,731	MTSAT	MSAS stations—Ibaraki, Australia
26	54.8	4,164	MTSAT	MSAS stations—Ibaraki, Australia, Hawaii
27	13.2	609	MTSAT	MSAS stations + Cold Bay
A		139	TEST	0 = 75'
B		422	TEST	0 = 30°
C		3,343	TEST	0 = 10'
D		13,211	TEST	0 = 5'
E		67	TEST	41 stations
F		64	TEST	41 + 4 stations

The four WAAS stations (CONUS) are Boston, Miami, Seattle, and Los Angeles.

The four WAAS stations (NE) are Boston, New York, Washington, DC, and Cleveland.

8.6.2 Joint Precision Approach and Landing System (JPALS)

The JPALS is being developed as an all-weather precision approach-and-landing system to meet DOD needs for aviation. The system is a LADGPS based system that will have various operational features to meet the wide variety of DOD requirements. These requirements include landing various fixed-wing and rotor-wing aircraft at civil land airfields as well as sea-based

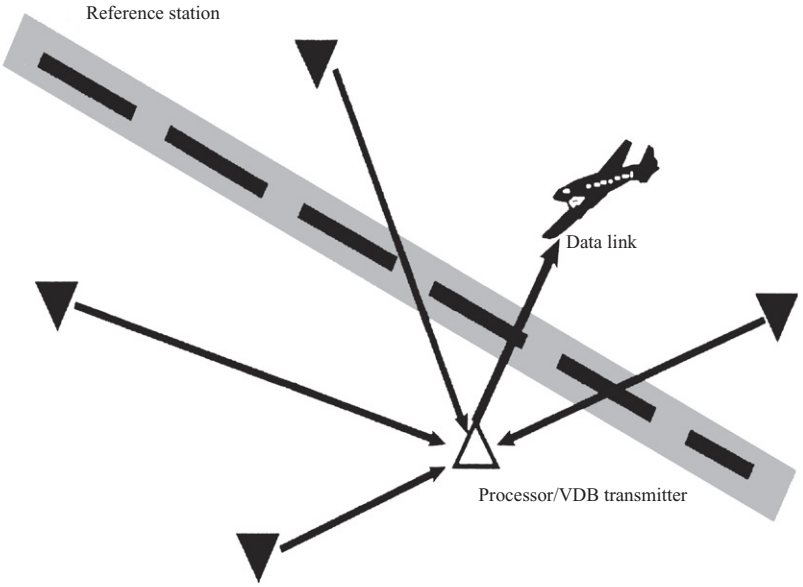


Fig. 8.10 Local-Area Augmentation System (LAAS). VDB, very high frequency data broadcast.

carrier platforms. The JPALS is planned to be interoperable with the civil land-based LAAS so that DOD platforms can be supported seamlessly by civil land bases. The JPALS user equipment features anti-jam capabilities, including the use of controlled reception pattern antennas (CRPAs) on some platforms. The JPALS utilizes both code and carrier phase based user solutions to support the various operational requirements. The JPALS program has accomplished numerous development and demonstration milestones and is scheduled for an initial operational capability in 2014 [20].

8.6.3 Enhanced Long-Range Navigation (eLoran)

While eLoran is not a GNSS and is not currently operational in the US, other parts of the world have been using eLoran as a backup to GNSS. eLoran was developed to enhance the capabilities provided by the legacy LORAN-C for data channel messaging, all-in-view, and traceable time reference to Coordinated Universal Time (UTC). Data messaging formats have been established within eLoran to provide differential eLoran capabilities [21, 22].

8.7 MEASUREMENT/RELATIVE-BASED DGNSS

DGNSS is a technique for improving the performance of GNSS positioning. The basic idea of DGNSS is to compute the spatial displacement vector of the user's receiver (sometimes called the roving or remote receiver) relative to another receiver (usually called the reference receiver or base station). In most DGNSS applications, the coordinates of the reference receiver are precisely known from highly accurate survey information; thus, the accurate location of the roving receiver can be determined by vector addition of the reference receiver coordinates and the reference-to-rover displacement vector.

The positioning accuracy of DGNSS depends on the error in estimating the reference-to-rover displacement vector. This error can be made considerably smaller than the positioning error of a stand-alone receiver because major components of pseudorange measurement errors are common to the roving and reference receivers and can be canceled out by using the difference between the reference and rover measurements to compute the displacement vector.

There are basically two ways that errors common to the roving and reference receiver can be canceled. The first method is called the *measurement* or *solution-domain* technique, in which both receivers individually compute their positions and the reference-to-rover displacement vector is simply the difference of these positions. However, the two receivers must use exactly the same set of satellites for this method to be effective. Since this requirement is often impossible to fulfill (e.g., due to blockage of signals at the roving receiver), this method is seldom used. A far better method, which offers more flexibility, is to use only the difference of the measurements from the set of satellites that are viewed in common by both receivers. Therefore, only this method will be described.

The two primary types of differential measurements are code measurements and carrier phase measurements.

8.7.1 Code Differential Measurements

To obtain code differential measurements, the roving and reference receivers each make a pseudorange measurement of the following form for each satellite:

$$\rho_M = \rho + cdt - cdT + d_{ION} + d_{TROP} + d_{EPHEM} + d_\rho, \tag{8.27}$$

where ρ_M is the measured pseudorange in meters, ρ is the true receiver-to-satellite geometric range in meters, c is the speed of light in meters per second, dt is the satellite clock error in seconds, dT is the receiver clock error in seconds, d_{ION} is the ionospheric delay error in meters, d_{TROP} is the tropospheric delay error in meters, d_{EPHEM} is the delay error in meters due to satellite ephemeris error, and d_ρ represents other pseudorange errors in meters, such

as multipath, interchannel receiver biases, thermal noise, and SA (when turned on). The pseudorange measurements made by both receivers must occur at a common GNSS time, or if not, corrections must be applied to extrapolate the measurements to a common time.

8.7.1.1 Single-Difference Observations A code single-difference observation is determined by subtracting an equation of the form (Eq. 8.27) for the reference receiver from a similar equation for the roving receiver, where both equations relate to the same satellite. The result is

$$\Delta\rho_{\mathrm{M}} = \Delta\rho - c\Delta dT + \Delta d_{\mathrm{ION}} + \Delta d_{\mathrm{TROP}} + \Delta d_{\mathrm{EPHEM}} + \Delta d_{\rho}, \tag{8.28}$$

where the symbol Δ denotes the difference between the corresponding terms in the two equations of the form (Eq. 8.27). Note that the term $c\Delta dT$ representing the satellite clock error has disappeared since the satellite clock error is the same for the pseudorange measurements made by each receiver. Furthermore, if the distance between the roving and reference receivers is sufficiently small (say, <20 km), the terms Δd_{ION}, Δd_{TROP}, and $\Delta d_{\mathrm{EPHEM}}$ will be nearly canceled out since errors due to the ionosphere, troposphere, and ephemeredes vary slowly with position.

8.7.1.2 Double-Difference Observations A code double-difference measurement is formed by subtraction of the single-difference observation of the form (Eq. 8.28) for one satellite from a similar single-difference observation for another satellite. Thus, if there are N single-difference observations corresponding to N satellites, there will be $N - 1$ independent double-difference observations that can be formed. The double-difference observations have the form

$$\nabla\Delta\rho_{\mathrm{M}} = \nabla\Delta\rho + \nabla\Delta d_{\mathrm{ION}} + \nabla\Delta d_{\mathrm{TROP}} + \nabla\Delta d_{\mathrm{EPHEM}} + \nabla\Delta d_{\rho} \tag{8.29}$$

where the symbol ∇ denotes difference between the corresponding difference terms in the two equations of the form (Eq. 8.28). Note that the double-difference error term $c\nabla\Delta dT$ involving receiver clock error has been cancelled out since receiver clock error is constant across all satellite measurements in both the reference and roving receivers. Furthermore, for a sufficiently small distance between the rover and reference receivers, the single-difference errors, Δd_{ION}, Δd_{TROP}, and $\Delta d_{\mathrm{EPHEM}}$, are so small that the corresponding double-difference errors $\nabla\Delta d_{\mathrm{ION}}$, $\nabla\Delta d_{\mathrm{TROP}}$, and $\nabla\Delta d_{\mathrm{EPHEM}}$ can be neglected. In this case, the double-difference observations become

$$\nabla\Delta\rho_{\mathrm{M}} \cong \nabla\Delta\rho + \nabla\Delta d_{\rho}, \tag{8.30}$$

which can result in positioning accuracies that are often in the submeter range.

Although DGPS is effective in removing satellite and receiver clock errors, ionospheric and tropospheric errors, and ephemeris errors, it cannot remove

errors due to multipath, receiver interchannel biases, and thermal noise since these errors are not common to the roving and reference receivers.

8.7.2 Carrier Phase Differential Measurements

Because carrier phase pseudorange measurements have significantly less noise than do those using the code, positioning accuracy is potentially much more accurate. However, since only the fractional and not the integer part of a carrier cycle can be observed, some method of finding the integer part must be employed. This is the classic *ambiguity resolution* problem.

Single- and double-difference observations can be obtained from carrier phase pseudorange measurements having the form

$$\lambda\phi_{\mathrm{M}} = \rho + cdt - cdT + \lambda N - d_{\mathrm{ION}} + d_{\mathrm{TROP}} + d_{\mathrm{EPHEM}} + d_{\phi}, \tag{8.31}$$

where the new variables are the carrier wavelength λ (0.1903 m for L1 and 0.2442 m for L2), the measured carrier phase ϕ_{M} in cycles, the carrier phase ambiguity N in cycles, and other errors d_{ϕ} in meters. Because the ionospheric group delay for the carrier is opposite that of the code, the ionospheric error is reversed in sign in Eq. 8.31.

In some cases, triple differences of carrier phase measurements are used, as will be subsequently described.

8.7.2.1 Single-Difference Observations Each carrier phase single-difference observation is determined in the same manner as for the code by subtracting an equation of the form (Eq. 8.31) for the reference receiver from a similar equation for the roving receiver, where both equations relate to the same satellite. The result is

$$\Delta\lambda\phi_{\mathrm{M}} = \Delta\rho - c\Delta dT + \lambda\Delta N - \Delta d_{\mathrm{ION}} + \Delta d_{\mathrm{TROP}} + \Delta d_{\mathrm{EPHEM}} + \Delta d_{\phi}, \tag{8.32}$$

where, as before, the satellite clock error term has disappeared and the terms Δd_{ION}, Δd_{TROP}, and $\Delta d_{\mathrm{EPHEM}}$ are small for small distances between the rover and reference receivers.

8.7.2.2 Double-Difference Observations A double-difference carrier phase measurement is formed by subtraction of the single-difference observation of the form (Eq. 8.32) for one satellite from a similar single-difference observation for another satellite. The double-difference observations have the form

$$\nabla\Delta\lambda\phi_{\mathrm{M}} = \nabla\Delta\rho + \lambda\nabla\Delta N - \nabla\Delta d_{\mathrm{ION}} + \nabla\Delta d_{\mathrm{TROP}} + \nabla\Delta d_{\mathrm{EPHEM}} + \nabla\Delta d_{\phi}. \tag{8.33}$$

Again, the receiver clock error term has disappeared, and the terms Δd_{ION}, Δd_{TROP}, and $\Delta d_{\mathrm{EPHEM}}$ are usually small. However, the value of N in the phase ambiguity term $\lambda\nabla\Delta N$ must be determined by some method of ambiguity resolution.

8.7.2.3 Triple-Difference Observations Triple-difference carrier observations are sometimes used in DGNSS to detect and correct cycle slips during carrier tracking. These observations have the form

$$\delta\nabla\Delta\lambda\phi_{\mathrm{M}} = \delta\nabla\Delta\rho + \delta\nabla\Delta d_{\mathrm{ION}} + \delta\nabla\Delta d_{\mathrm{TROP}} + \delta\nabla\Delta d_{\mathrm{EPHEM}} + \delta\nabla\Delta d_{\phi}, \quad (8.34)$$

where δ is the time difference between two successive double-difference observations. Cycle slips can be detected by observing the deviation of successive triple difference observations from their predicted values as the carrier is tracked [23, 24].

8.7.2.4 Combinations of L1 and L2 Carrier Phase Observations Summing the L1 and L2 double-difference carrier observations results in higher-resolution phase measurements than can be obtained at either frequency alone. Such narrow-lane measurements result in more precision but place greater demands on phase ambiguity resolution. On the other hand, it is easier to resolve the phase ambiguity of wide-lane measurements obtained by differencing the L1 and L2 observations at the expense of reduced resolution.

8.7.3 Positioning Using Double-Difference Measurements

8.7.3.1 Code-Based Positioning The linearized matrix equation for positioning using code double-difference measurements from four satellites has the form

$$\overbrace{\delta Z_{\nabla\Delta\rho}}^{3\times1} = \overbrace{H^{[1]}_{\nabla\Delta\rho}}^{3\times3}\overbrace{\delta\mathbf{x}}^{3\times1} + \overbrace{\mathbf{v}_{\rho}}^{3\times1}, \quad (8.35)$$

which is the same form as shown in Section 2.2.3. However, because the double-difference measurements have eliminated receiver clock error as an unknown, the unknowns are simply the X, Y, and Z coordinates of the roving receiver, constituting the components of the 3×1 vector $\delta\mathbf{x}$. Thus, the measurement matrix $H^{[1]}$ is 3×3 and the partial derivatives in it are partial derivatives of the double differences $\nabla\Delta\rho_{\mathrm{M}}$ with respect to user position coordinates X, Y, Z instead of partial derivatives of the pseudorange measurements. Accordingly, the measurement vector $\delta Z\rho$ and the measurement noise vector $\mathbf{v}_{\rho}$ are 3×1. As indicated in Section 2.2.3, a solution for position can be found by computing the measurement vector associated with an assumed initial position $\mathbf{x}$, finding the difference δZ_{ρ} between the computed and actual measurement vectors, solving (Eq. 8.35) (omitting the measurement noise vector) for the position correction δx, and obtaining the new value $x + \delta\mathbf{x}$ for X. Iteration of this process is used to produce a sequence of positions that converges to the position solution.

8.7.3.2 Carrier Phase-Based Positioning For positioning using carrier phase double-difference measurements, the linearized matrix equation from four satellites used for iterative position solution has the form

$$\overbrace{\delta Z_{\nabla\Delta\phi}}^{3\times1} = \overbrace{H^{[1]}_{\nabla\Delta\phi}}^{3\times3}\overbrace{\delta\mathbf{x}}^{3\times1} + \overbrace{\mathbf{v}_\rho}^{3\times1}, \tag{8.36}$$

where the measurement matrix $H^{[1]}$ contains the partial derivatives of the double differences $\nabla\Delta\lambda\phi_{\mathrm{M}}$ with respect to user position coordinates X, Y, Z. As compared to code-based positioning, the measurement noise term $\mathbf{v}_\rho$ is much smaller, often in the centimeter range. However, the major difference is that the ambiguity in the phase measurements can cause convergence to any one of many possible positions in a spatial grid of points. Only one of these points is the correct position. Various techniques for resolving the ambiguity have been developed. A simple method is to use the position solution from the code double-difference measurements as the initial position X in the carrier phase iterative position solution. If this initial position is sufficiently accurate, convergence to the correct solution will be obtained.

8.7.3.3 Real-Time Processing versus Postprocessing Since double differencing combines measurements made in the roving and reference receivers, these measurements must be brought together for processing. Often the processing site is at the roving receiver, although in other applications it can be at the reference station or at another off-site location. In real-time processing, measurements are transmitted to the processing site using wireless communication or a telephone link. In postprocessing, the data can be physically carried to the processing site in a storage medium such as a floppy disk or a CD-ROM. Another postprocessing option is to transmit the data via the Internet.

8.8 GNSS PRECISE POINT POSITIONING SERVICES AND PRODUCTS

The cost and inconvenience of setting up one's own DGNSS system can be eliminated because there are numerous services and software packages available to the user, some of which are free. There are too many to describe completely, so only a few of them are described in this section.

8.8.1 The International GNSS Service (IGS)

Many of the DGNSS services are subsumed under the IGS, which is a voluntary federation of more than 200 worldwide agencies that pool resources and permanent GPS and Global Orbiting Navigation Satellite System (GLONASS) station data to generate precise DGNSS positioning services. The IGS is committed to providing the highest-quality data and products as the standard for global navigation satellite systems (GNSSs) in support of earth science research, multidisciplinary applications, and education. The IGS also intends to incorporate future GNSS systems, such as Galileo, as they become operational.

8.8.2 Continuously Operating Reference Stations (CORSs)

The National Geodetic Survey (NGS), an office of NOAA's National Oceanic and Atmospheric Administration (NOAA) National Ocean Service, manages two networks of CORS: the National CORS network and the Cooperative CORS network. These networks consist of numerous base stations containing DGPS reference receivers that operate continuously to generate pseudorange and other DGPS data for postprocessing. The data are disseminated to a wide variety of users. Surveyors, geographic information system (GIS)/Land Information System (LIS) professionals, engineers, scientists, and others can apply CORS data to their own GPS measurements to obtain positioning accuracies approaching a few centimeters relative to the National Spatial Reference System (NSRS), both horizontally and vertically. The CORS program is a multipurpose cooperative endeavor involving more than 130 government, academic, and private organizations, each of which operates at least one CORS site. In particular, it includes all existing National Differential GPS (NDGPS) sites and all existing FAA WAAS sites. New sites are continually being evaluated according to established criteria.

Typical uses of CORS include land management, coastal monitoring, civil engineering, boundary determination, mapping and GISs, geophysical and infrastructure modeling, as well as future improvements to weather prediction and climate modeling.

All national CORS data are available from NGS at their original sampling rate for 30 days, after which the data are decimated to a 30-s sampling rate. Cooperative CORS data are available from a large number of participating organizations that operate individual sites. Most of the CORS data are available on the Internet.

8.8.3 GPS Inferred Positioning System (GIPSY) and Orbit Analysis Simulation Software (OASIS)

The GIPSY-OASIS H (GOA II) package consists of extremely versatile software that can be used for GPS positioning and satellite orbit analysis. Developed by the Caltech Jet Propulsion Laboratory (WE), it can provide centimeter-level DGPS positioning accuracy over short to intercontinental baselines. It is capable of unattended, automated, low-cost operation in near real time for precise positioning and time transfer in ground, sea, air, and space applications.

GOA II also includes many force models useful for OD, such as earth/sun/moon/planet (and tidal) gravity perturbations, solar pressure, thermal radiation, and drag, which make it useful in non-GPS satellite positioning applications. To augment its potential accuracy, models are included for earth characteristics, such as tides, ocean/atmospheric loading, and crustal plate motion.

Parameter estimation for positioning and time transfer is state of the art. A general estimator can be used for GPS and non-GPS data. Matrix factorization is used to maintain robustness of solutions, and the estimator can intelligently identify, correct, or exclude questionable data. A general and flexible noise model is included.

8.8.4 Australia's Online GPS Processing System (AUPOS)

AUPOS provides users with the facility to submit via the Internet dual-frequency geodetic quality GPS RINEX data observed in a "static" mode and receive rapid-turnaround precise position coordinates. The service is free and provides both International Terrestrial Reference Frame (ITRF) and Geocentric Datum of Australia (GDA94) coordinates. This Internet service takes advantage of both IGS products and the IGS GPS network and can handle GPS data collected anywhere on earth.

8.8.5 Scripps Coordinate Update Tool (SCOUT)

SCOUT, managed by the Scripps Institute of Oceanography, is also a system that provides precise positioning for users who submit GPS RINEX data from their receiver via the Internet. The reference stations are by default the three nearest sites for which data have been collected and are available for the specific day the user's data are taken. However, the user can specify the reference stations if desired. Station maps are provided to assist the user in specifying nearby reference sites. When SCOUT has finished determining a DGPS position solution, it sends a report of the results to the user via the Internet. The report contains both Cartesian and geodetic coordinates, standard deviations, and the locations of the reference sites that were used. The reported Cartesian coordinates are referenced to the International Terrestrial Reference Frame 2000 (ITRF2000), and the geodetic coordinates are referenced to both ITRF2000 and the World Geodetic System 1984 (WGS84) ellipsoid.

8.8.6 The Online Positioning User Service (OPUS)

The NGS operates OPUS as a means to provide GPS users easier access to the NSRS. OPUS users submit their GPS data files to the NGS Internet site. The NGS computers and software determine a position by using reference receivers from three CORS sites. The position is reported back to the user by e-mail in both ITRF and North American Datum 1983 (NAD83) coordinates, as well as Universal Transverse Mercator (UTM), and State Plain Coordinate (SPC) northing and easting. Results are typically obtained within a few minutes. OPUS is intended for use in the coterminous United States and in most U.S. territories. It is NGS policy not to publish geodetic coordinates outside the United States without the agreement of the affected countries.

PROBLEMS

8.1 Determine the CCC at the GUS location using L1 code and carrier.

8.2 Determine the frequency stability of the GEO transponder using Allan variance for the L1 using 1- to 10-s intervals.

8.3 What are GNSS

(a) single difference?

(b) double difference?

(c) triple difference?

(d) wide lane?

(e) narrow lane?

8.4 What is a CORS site?

REFERENCES

[1] Institute of Navigation, *Global Positioning System, Selected Papers on Satellite Based Augmentation Systems (SBASs)* ("Redbook"), Vol. VI. ION, Alexandria, VA, 1999.

[2] R. B. Langley, "A GPS Glossary," *GPS World*, October 1995, pp. 61–63.

[3] L. A. Cheung and P. H. Shu, "WAAS Single Frequency GEO Operation Field Test and L1 Signal Code-Carrier coherence," *ION GNSS 2011 Conference Proceedings*, September 20–23, 2011.

[4] M. S. Grewal, "Space-Based Augmentation for Global Navigation Satellite Systems," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* **59**(3): 497–504 (March 2012).

[5] S. Peck, C. Griffith, V. Reinhardt, W. Bertiger, B. Haines, and G. M. R. Winkler, "WAAS Network Time Performance and Validation Results," *Proceedings of the Institute of Navigation* (Santa Monica, CA), ION, Alexandria VA, January 1998.

[6] R. Ahmadi, G. S. Becker, S. R. Peck, F. Choquette, T. F. Gerard, A. J. Mannucci, B. A. Iijima, and A. W. Moore, "Validation Analysis of the WAAS GIVE and UIVE Algorithms," *Proceedings of the Institute of Navigation*, ION '98 (Santa Monica, CA), ION, Alexandria, VA, January 1998.

[7] M. S. Grewal, W. Brown, and R. Lucy, "Test Results of Geostationary Satellite (GEO) Uplink Sub-System (GUS) Using GEO Navigation Payloads," in *Monographs of the Global Positioning System: Papers Published in Navigation ("Redbook")*, Vol. VI. Institute of Navigation, ION, Alexandria, VA, 1999, pp. 339–348.

[8] B. W. Parkinson and J. J. Spilker, Jr. (Eds.), *Global Positioning System: Theory and Applications*, Vol. 1, Progress in Astronautics and Aeronautics (series). American Institute of Aeronautics and Astronautics, Washington, DC, 1996.

[9] B. W. Parkinson and J. J. Spilker, Jr. (Eds.), *Global Positioning System: Theory and Applications*, Vol. 2, Progress in Astronautics and Aeronautics (series). American Institute of Aeronautics and Astronautics, Washington, DC, 1996.

[10] B. W. Parkinson, M. L. O'Connor, and K. T. Fitzgibbon, "Aircraft Automatic Approach and Landing Using GPS," Chapter 14 in B. W. Parkinson and J. J. Spilker, Jr. (Eds.), *Global Positioning System: Theory & Applications*, Vol. II, Progress in Astronautics and Aeronautics (series), Vol. 164, Paul Zarchan editor-

in-chief. American Institute of Aeronautics and Astronautics, Washington, DC, 1995, pp. 397–425.

[11] T. Yunk, et al., A Robust and Efficient New Approach to Real Time Wide Area Differential GPS Navigation for Civil Aviation, NASA/JPL Internal Report JPL 1)-12584, 1995, Pasadena, CA.

[12] M. S. Grewal, P. Hsu, and T. W. Plummer, "A New Algorithm for SBAS GEO Uplink Subsystem (GUS) Clock Steering," *ION GPS/GNSS Proceedings*, September 2003, pp. 2712–2719.

[13] RTCA, *Minimum Operational Performance Standards (MOPS) for Global Positioning System/Wide Area Augmentation System Airborne Equipment*, RTCA/DO-229, January 16, 1996, and subsequent changes, Appendix A, "WAAS System Signal Specification," RTCA, Washington, DC.

[14] D. W. Allan, *The Measurement of Frequency and Frequency Stability of Precision Oscillators*, NBS Technical Note 669. 1975, pp. 1–27.

[15] E. N. Carolipio, N. Pandya, and M. S. Grewal, "GEO Orbit Determination via Covariance Analysis with a Known Clock Error," *Navigation, Journal of the Institute of Navigation* **48**(4), 255–260 (2002).

[16] P. Tran and J. DiLellio, "Impacts of GEOs as Ranging Sources on Precision Approach Category I Availability," *Proceedings of ION Annual Meeting*, Institute of Navigation, Alexandria, VA, June 2000.

[17] M. S. Grewal, N. Pandya, J. Wu, and E. Carolipio, "Dependence of User Differential Ranging Error (UDRE) on Augmentation Systems—Ground Station Geometries," *Proceedings of the Institute of Navigation's (ION) 2000 National Technical Meeting* (Anaheim, CA), January 26–28, 2000, ION Alexandria, VA, 2000, pp. 80–91.

[18] N. Pandya, "Dependence of GEO UDRE on Ground Station Geometries," in WAAS *Engineering Notebook*, Raytheon Systems Company, Fullerton, CA, December 1, 1999.

[19] R. R. Bate, D. D. Mueller, and J. E. White, *Fundamentals of Astrodynamics*. Dover, New York, 1971.

[20] Naval Air Systems Command, Joint Precision Approach & Landing System, http://www.navair.navy.mil/index.cfm?fuseaction=home.display&key=55F82859-4F70-434F-9A2A-9E65448AF52A, accessed November 21, 2012.

[21] General Lighthouse Authority, Research & Radionavigation, http://www.glarrnav.org/radionavigation/eloran/index.html, visited November 21, 2012.

[22] International Loran Associations, Enhanced Loran (eLoran) Definition Document, Report Version 1.0, Report Version Date: 16 October 2007, http://www.loran.org/ILAArchive/eLoran%20Definition%20Document/eLoran%20Definition%20Document-1.0.pdf, visited November 21, 2012.

[23] S. Kiran and C. Bartone, "A Viable Airport Pseudolite Architecture for the Local Area Augmentation System," *Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003)* (Portland, OR), September, 2003, pp. 2326–2336.

[24] F. van Graas and A. Soloviev, "Precise Velocity Estimation Using a Stand-Alone GPS Receiver," *NAVIGATION, Journal of the Institute of Navigation*, **51**(4): 283–292.

9

GNSS AND GEO SIGNAL INTEGRITY

9.1 INTRODUCTION

Navigation system integrity refers to the ability of the system to provide timely warnings to users when the system should not be used for navigation. Global navigation satellite systems (GNSSs) have both internal and independent methods to maintain integrity. Satellites monitor for some of the anomalies, but not all. Clock failures, data errors, selective availability (SA, currently discontinued), and antispoof (AS) are checked internally. The master control station monitors the constellations. In the case of Global Positioning System (GPS), data are collected from five monitoring stations distributed around the earth. GPS performance is checked every 15 min by conducting tolerance and validation checks of the measured pseudoranges, using a Kalman filter, error management process [1].

The basic GNSS (as described in Chapter 4) provides integrity information to the user via the navigation message, but this may not be timely enough for some applications, such as civil aviation. Therefore, additional methods of providing integrity are necessary.

Two different methods will be discussed—GNSS-only receiver (TSO-C129-compliant) autonomous integrity monitoring (RAIM) and use of ground monitoring stations to monitor the health of the satellites, as is done via space-

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

based augmentation system (SBAS) and ground-based augmentation system (GBAS) (TSO-C145-compliant receivers).

The analytic structure of RAIM is stochastic detection theory. Two hypothesis testing questions are raised. First, has the failure occurred? Second, if so, which satellite failed? In the case of no backup navigation system, both questions must be answered. The bad satellite must be identified and eliminated from the navigation solution, so that the vehicle can proceed safely without the bad GNSS solution. However, if there is a backup navigation system available, then it can be used when a failure occurs.

Determining which satellite has failed is more difficult than failure detection, and it requires more measurement redundancy [2–5].

Three RAIM methods have been proposed in recent papers on GPS integrity:

1. range comparison method
2. least-squares residual method
3. parity method.

The three methods are called "snapshot methods" because the detection algorithms assume that noisy redundant range-type measurements are available at a given point in time. The basic measurement equation is derived in Chapter 2, Eq. 2.31. The following measurement equation is used in all three methods:

$$\overbrace{\delta Z_\rho}^{n\times 1} = \overbrace{H}^{n\times 4}\overbrace{\delta \mathbf{x}}^{4\times 1} + \overbrace{v_\rho}^{4\times 1},$$

where n is the number of satellites and H is the linearized sensitivity matrix about nominal user position and clock bias.

9.1.1 Range Comparison Method

For the GNSS navigation problem described in Chapter 2, Section 2.2.3, there are four unknowns (three position coordinates $[X, Y, Z]$ and clock bias C_b) and more than four satellites in view (e.g., six satellites). One can solve the position and time equations for the first four satellites, ignoring noise, and find the user position. This solution can then be used to predict the remaining two pseudorange measurements, and the predicted values could be compared with actual measured values. If the two differences (residuals) are small, we have near consistency in the measurements and the detection algorithm can declare "no failure." It only remains to quantify what we mean by "small" or "large" and then assess the decision rule performance on actual data.

There are six satellites in view. With the range comparison method, two range residuals ($\tilde{\delta}Z_\rho^1$, $\tilde{\delta}Z_\rho^2$) represent a point in a statistical plane as shown in Fig. 9.1.

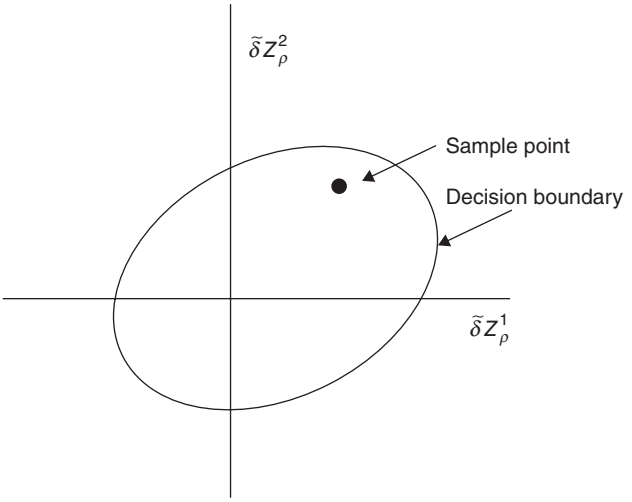


Fig. 9.1 Test statistic plane for the six satellites in view.

If the statistics of the noise (v_ρ) are Gaussian (normal), the contour will be elliptical as shown in Fig. 9.1. The particular contour chosen is the one that sets the alarm rate at the desired value. The alarm rate could be set at 1/15,000 as specified in the RTCA MOPS [6, 7].

9.1.2 Least-Squares Method

The basic measurement equation with noise (Eq. 2.31 from Chapter 2) is

$$\delta Z_\rho = \mathbf{H}\delta\mathbf{x} + v_\rho, \tag{9.1}$$

where the additive white noise $v_\rho \in N(0, \sigma^2)$.

Let us suppose six satellites are in view and four unknowns, as in Section 9.1.1, and solve for the four unknowns by the least-squares method.

The least-squares solution is given by Eq. 2.34:

$$\widehat{\delta\mathbf{x}} = \left(\mathbf{H}^{\mathrm{T}}\mathbf{H}\right)^{-1}\mathbf{H}^{\mathrm{T}}\delta Z_\rho. \tag{9.2}$$

The least-squares solution can be used to predict the six measurements, in accordance with

$$\widehat{\delta Z_\rho}\,(\text{predicted}) = \mathbf{H}\widehat{\delta\mathbf{x}}. \tag{9.3}$$

We can get a formula for the sum-squared residual error S by substituting $\widehat{\delta\mathbf{x}}$ from Eq. 9.2 into Eq. 9.3:

$$\Delta Z_\rho = \delta Z_\rho - \widehat{\delta Z_\rho} (\text{residual error}) \tag{9.4}$$

$$= \left[\mathbf{I} - \mathbf{H} \left(\mathbf{H}^{\mathrm{T}} \mathbf{H} \right)^{-1} \mathbf{H}^{\mathrm{T}} \right] \delta Z_\rho \tag{9.5}$$

$$S = \Delta Z_\rho^T \Delta Z_\rho, \text{ the sum-squared error.} \tag{9.6}$$

This sum of squared error has three properties that are important in the decision rule:

1. S is a nonnegative scalar quantity. Choose a threshold value τ of S such that $S < \tau$ will be considered safe and that $S \geq \tau$ will be declared a failure.
2. If the v_ρ have the same independent zero-mean Gaussian distribution, then the statistical distribution of S is completely independent of the satellite geometry for any number of satellites (n). Thresholds are precalculated, which results in the desired alarm rate for the various anticipated values of n. Then the real-time algorithm sets the threshold appropriately for the number of satellites in view at the moment.
3. With the v_ρ, from above, S has an unnormalized chi-square (χ^2) distribution with $(n - 4)$ degrees of freedom (see Chapter 10, Section 10.9.4.1). Parkinson and Axelrad [2] use $\sqrt{S/n - 4}$ as the test statistic. Calculating the test statistic involves the same matrix manipulation, but these are no worse than calculating the dilution of precision (DOP) [3].

9.1.3 Parity Method

The parity RAIM method is somewhat similar to the range comparison method except that the way in which the test statistic is formed is different. In the parity method, perform a linear transformation on the measurement vector as follows:

$$\begin{bmatrix} \delta \mathbf{x} \\ p \end{bmatrix} = \begin{bmatrix} \left(\mathbf{H}^{\mathrm{T}} \mathbf{H} \right)^{-1} \mathbf{H}^{\mathrm{T}} \\ \mathbf{P} \end{bmatrix} \delta Z_\rho. \tag{9.7}$$

The lower portion of Eq. 9.7, which yields p, is the result of operating on δZ_ρ with the special $(n - 4) \times n$ matrix $\mathbf{p}$, whose rows are mutually orthogonal, unity magnitude and orthogonal to the columns of $\mathbf{H}$.

Under the same assumptions about the noise v_ρ as above, the following statements can be made:

$$\left.\begin{aligned} E\langle p \rangle &= 0 \\ E\left\langle pp^T \right\rangle &= \sigma^2 \mathbf{I} \, (\text{covariance of } p) \end{aligned}\right\}, \tag{9.8}$$

where σ^2 is the variance associated with v_ρ. Use p as the test statistic in this method. For detection, obtain all the information needed about p from its

magnitude or magnitude squared. Thus, in the parity method, the test statistic for detection reduces to a scalar, as in the least-squares method [8, 9].

These RAIM protection levels have assumed that there is at most one ranging source with bias. Extending RAIM protection to multiple bias is given in Refs. 10–12.

9.2 SBAS AND GBAS INTEGRITY DESIGN

The objectives of SBAS and the GBAS are to provide integrity, accuracy, availability and continuity for GPS, Global Orbiting Navigation Satellite System (GLONASS), and Galileo Standard Positioning Service (SPS). Integrity is defined as the ability of the system to provide timely warnings to the user when individual corrections or certain satellites should not be used for navigation, specifically, the prevention of hazardously misleading information (HMI) data transmission to the user. The system should not be used for navigation when hardware, software, or environmental errors directly pose a threat to the user or indirectly pose a threat by obscuring HMI from the integrity monitors. SBAS integrity is based on the premise that errors not detected or corrected in the operational environment can become threats to integrity and, if not mitigated, can become hazards to the user.

An SBAS design should mitigate the majority of these data errors with corrections that are proved to bound the integrity hazard to an acceptable level. The leftover data errors (referred to as *residual errors*) are mitigated by the transmission of residual error bounding information. The threat of potential underbounding of integrity information is mitigated by integrity monitors. This section examines both the faulted and unfaulted cases and mitigation strategies for these cases. These SBAS corrections improve the accuracy of satellite signals. The integrity data ensure that the residual errors are bounded. The SBAS integrity monitors help ensure that the integrity data have not been corrupted by SBAS failures.

The section addresses the data errors, error detection and correction pitfalls, and how such threats can become HMI to the user, as well as fault conditions, failure conditions, threats, and mitigation, and how safety integrity requirements are satisfied. Safety integrity assurance rules will be evaluated. Results from real signal in space (SIS) data, a high-level overview of the required SBAS safety architecture, and a data processing path protection approach are included.

This section provides information that defines how a safety-of-life-critical SBAS system should be designed and implemented in order to ensure mitigation of the entire International Civil Aviation Organization (ICAO) threat space to the required level less than 10^{-7}. It provides as an example, the rationale, background, and references to show that the SBAS can be used as a trusted navigational aid to augment the GPS for lateral positioning with vertical guidance (LPV).

Rail integrity is one of the most stringent operational requirements, as evidenced by the European Rail Traffic Management System (ERTMS) required integrity levels, which are in the order of 10^{-11}. Train detection will require an equally high level of positive integrity.

The section addresses the hazardous/severe–major integrity failure condition using LPV as an illustrative example. The ICAO integrity requirement is based on the premise that errors not detected or corrected in the operational environment system can become threats to the integrity and, if not mitigated, can become hazards to the user. These errors in the operational environment (referred to as data errors) can affect both the user and the SBAS system. Integrity in this context is defined as the ability of the system to provide timely warning to users when individual corrections or satellites should not be used for navigation, that is, the prevention of HMI data transmission to the user. The system should not be used for navigation when data errors in the environment, such as the ionosphere, and data processing, such as multipath, render the integrity data erroneous. The user must be protected from residual errors that can become threats to the integrity data that could result in HMI being transmitted to the user [13–15].

An SBAS design mitigates the majority of these data errors with "corrections." The leftover data errors (referred to as residual errors) are mitigated by the transmission of residual error bounding information. The threat of potential underbounding of the integrity information is mitigated by integrity monitors and point design features that protect the integrity of the information within the SBAS system. Additionally, analytic safety analyses are required to provide evidence and proof that the residual errors are acceptable (i.e., that the probability of HMI transmission to the user is sufficiently low).

Table 9.1 lists the SBAS error sources. Mitigation of these errors when they become integrity threats are presented in Section 9.2.8. Section 9.3 gives an application of these techniques to SBAS for threat mitigation. GNSS Integrity Channel (GIC) is discussed in Section 9.5.

9.2.1 SBAS Error Sources and Integrity Threats

The SBAS operational environment contains data errors. The SBAS ensures that these data errors do not become threats to the integrity data, so that HMI is not broadcasted to the user with a P_{HMI} greater than 10^{-7}.

TABLE 9.1. List of SBAS Error Sources

GNSS satellite	Integrity bound associated
GEO satellites	Message uplink
Reference receiver	Environment (ionosphere and troposphere)
Estimation	

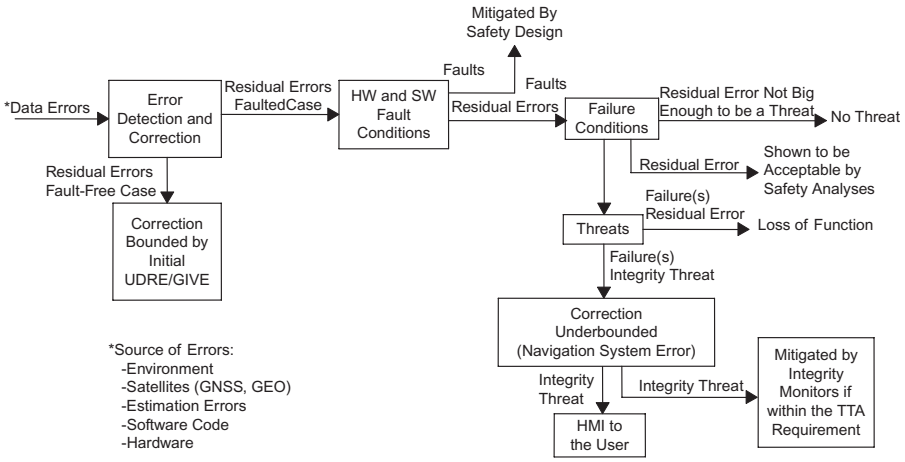


Fig. 9.2 Integrity mitigation within an SBAS.

The data used by an SBAS to calculate the correction and/or integrity data are assumed to contain errors, such as GNSS satellite clock offset, which must be sufficiently mitigated. The errors discussed are inherent in any SBAS design that utilizes GPS, Galileo, GLONASS, or geostationary earth orbit (GEO) satellites; reference receivers; corrections; and integrity bounds. Depending on the system architecture, other error sources may also exist. Table 9.1 summarizes the error sources that every SBAS system must address [16].

The integrity threats associated with each of these error sources generally have two cases, shown in Fig. 9.2. The *fault-free case* addresses the nominal errors associated with each error source and the *faulted* case represents the errors when one or more of the system's components cause errors. The defining quality of an SBAS system that meets the ICAO standards is the mitigation of the faulted case and the fault-free case.

9.2.2 GNSS-Associated Errors

GNSS error sources are mitigated in an SBAS system by using corrections and integrity bounds. Generally, the SBAS system corrects the errors as well as possible and then bounds the residual errors with integrity bounds that are broadcast to the user. The nominal GNSS satellite errors are well understood. The literature includes many techniques for mitigating these errors. GNSS failure modes are not as well understood and often require careful study to define the threat, which must be accounted for in threat models.

9.2.2.1 GNSS Clock Error Each GNSS satellite broadcasts a navigation data message containing an estimate of its clock offset (relative to GNSS time) and drift rate. The GNSS satellite clock value is utilized to correct the satellite's

pseudorange, the measurement used to calculate the distance (range) from the satellite to the receiver (either the user's receiver or the reference receiver).

Under fault-free conditions, the SBAS can accurately compute these corrections and mitigate this error source. Simple statistical techniques can be used to characterize these errors. The SBAS must also address satellite failures that cause the clock to rapidly accelerate, rendering the corrections suddenly invalid. As a result, the error bounds may not be bounding the residual error in the corrections. These types of failures have been observed many times in the history of GNSS.

9.2.2.2 GNSS Ephemeris Error Each GNSS also broadcasts a navigation data message containing a prediction of its orbital parameters—Keplerian orbital parameters. The satellite's ephemeris data enable determination of the satellite position and velocity. Any difference between the satellite's calculated position and velocity and the true position is a potential source of error.

Under fault-free conditions, the SBAS provides corrections relative to the GNSS broadcast ephemeris data. The SBAS can accurately compute these corrections and mitigate this error source using standard statistical techniques. The satellite may experience an unexpected maneuver, rendering the corrections suddenly invalid. This threat includes geometric constraints that may be insufficient for the SBAS to adequately detect the orbit error.

Under fault-free conditions, the SBAS provides corrections relative to the GNSS broadcast ephemeris data. The SBAS can accurately compute these corrections and mitigate this error source using standard statistical techniques. The satellite may experience an unexpected maneuver, rendering the corrections suddenly invalid. This threat includes geometric constraints that may be insufficient for the SBAS to adequately detect the orbit error.

9.2.2.3 GNSS Code and Carrier Incoherence The GNSS signal consists of a radiofrequency carrier encoded with a pseudorandom spread-spectrum code. The user's receiver performs smoothing of its pseudorange measurements using the carrier phase measurements. If the code and carrier are not coherent, there will be an error in this pseudorange smoothing process. This error is caused by a satellite failure. Incoherence between the code and carrier phase can increase the range error, ultimately resulting in the user incorrectly determining the code/carrier ambiguity.

9.2.2.4 GNSS Signal Distortion A satellite may fail in a manner that distorts the pseudorange portion (PRN encoding) of the GPS transmission. This causes an error in the user's pseudorange measurements that may be different from the error that the SBAS receiver experiences. In 1993, the SV-19 GPS satellite experienced a failure that fits into this category. This error is caused by a satellite failure. If a satellite experiences this type of failure, the SBAS may not be able to estimate the satellite clock corrections that are aligned with the user's measurements, which could result in HMI.

9.2.2.5 GNSS L1L2 Bias The GNSS L1 and L2 signals are utilized together to compute the ionospheric delay so the delay can be removed from the range calculations. The satellite has separate signal paths for these two frequencies; therefore, the signals can have different delays. The difference in the delays must be modeled accurately to be able to properly calibrate and use L1 and L2 signals together.

Under nominal conditions, the SBAS estimation process is very accurate, and this error is easily modeled with standard statistical techniques. If a satellite experiences a fault, the L1L2 bias can suddenly change, resulting in a large estimation error. A large estimation error can lead to excessive errors in correction processing.

9.2.2.6 Environment Errors: Ionosphere As the GNSS L1 and L2 signals propagate through the ionosphere, the signals are delayed by charged particles. The density of the charged particles, and therefore the delay, varies with location, time of day, angle of transmission through the ionosphere, and solar activity. This delay will cause an error in range measurements and must be corrected and properly accounted for in the SBAS measurement error models. As discussed earlier in Chapter 7, during calm ionospheric conditions, modeling errors are well understood and can be handled using standard statistical techniques. Ionospheric storms pose a multitude of threats for SBAS users. The model used in the error estimation may become invalid. The user may experience errors that are not observable to the SBAS due to the geometry of the reference station pierce points. The error in the corrections may increase over time due to rapid fluctuations in the ionosphere.

9.2.2.7 Environment Errors: Troposphere As the GNSS L1 and L2 signals propagate through the troposphere, the signals are delayed. This delay is dependent on temperature, humidity, angle of transmission through the atmosphere, and atmospheric pressure. This delay will cause an error in range measurements and must be corrected and properly accounted for in the measurement error models. Tropospheric modeling errors manifest themselves in the algorithms that generate the corrections. The user utilizes a separate tropospheric model that may have errors due to tropospheric modeling.

9.2.3 GEO-Associated Errors

9.2.3.1 GEO Code and Carrier Incoherence The GEO signal consists of a radiofrequency carrier encoded with a pseudorandom spread-spectrum code. The user's receiver performs smoothing of its GEO pseudorange measurements using the carrier phase measurements. If the code and carrier are not coherent, there will be an error in this pseudorange smoothing process. Under fault-free conditions, some incoherence is possible (due to environmental effects). This will be a very small error that is easily modeled by the ground

system. Under faulted conditions, severe divergence and potentially large errors are theoretically possible if the GEO uplink subsystem fails.

9.2.3.2 GEO-Associated Environment Errors: Ionosphere As the GEO L1 signal propagates through the ionosphere, the signal is delayed by charged particles. The density of the charged particles, and therefore the delay, varies with location, time of day, angle of transmission through the ionosphere, and solar activity. GEO satellites that are available today broadcast single-frequency (L1) signals that do not allow a precise determination of the ionospheric delay at a reference station. Without dual-frequency measurements, uncertainty in the calculated ionospheric delay estimates bleed into the corrections. New GEO satellites [PRN 135,138] have two frequencies L1 and L5. Ionospheric delay is calculated using those frequencies (see Chapter 7).

9.2.3.3 GEO-Associated Environment Errors: Troposphere Like GNSS satellite signals, the GEO L1 signal is delayed as it propagates through the troposphere. This delay will cause an error in range measurements and must be corrected and properly accounted for in the measurement error models.

9.2.4 Receiver and Measurement Processing Errors

Measurement errors affect an SBAS system in two ways. They can corrupt or degrade the accuracy of the corrections. They can also mask other system errors and result in HMI slipping through to the user. The errors given below must all be mitigated and residual errors bounded.

9.2.4.1 Receiver Measurement Error The receiver outputs pseudorange and carrier phase measurements for all satellites that are in view. The receiver and antenna characteristics limit the measurement accuracy. Under fault-free conditions, these errors can be addressed using well-documented processes. A receiver could fault and output measurement data that are in error for any or all of the satellites in view. A latent common-mode failure in the receiver firmware could cause all measurements in the system to simultaneously fail. Erroneous measurements pose two threats. They cannot only result in correction errors; they can also fool the integrity monitors and let HMI slip through to the user.

9.2.4.2 Intercard Bias For receiver designs that include multiple correlators, the internal delays in the subreceivers are different. This creates a different apparent clock for each subreceiver, called an *intercard bias*. Under nominal conditions, the intercard bias estimate is extremely accurate and the intercard bias error is easily accounted for. Any failure condition in the receiver or the algorithm computing the bias will result in an increase in the measurement data error.

9.2.4.3 Multipath Under nominal conditions, the dominant source of noise is multipath. Multipath is caused by reflected signals arriving at the receiver delayed relative to the direct signal. The amount of error is dependent on the delay time and the receiver correlator type. (See discussion of *multipath mitigation methods* in Chapter 7.)

9.2.4.4 L1L2 Bias The GNSS L1 and L2 signals are utilized together to compute the ionospheric delay so that the delay can be removed from the range calculations. The receivers and antenna will experience different delays in the electronics when monitoring these two frequencies. The difference in the delays must be accurately modeled to be able to remove the bias and to use the L1 and L2 signals together. If a receiver fails, the L1L2 bias can suddenly change, resulting in large estimation errors. Under nominal conditions, the estimation process is very accurate and the error is not significant.

9.2.4.5 Receiver Clock Error A high-quality receiver generally utilizes a (cesium) frequency standard that provides a long-term stable time reference (clock). This clock does drift. If a receiver fails, the clock bias can suddenly change, resulting in a large estimation error. Under nominal conditions, the SBAS is able to accurately account for receiver clock bias and drift. If a receiver fails, the clock may accelerate, introducing errors into the corrections and the integrity monitoring algorithms.

9.2.4.6 Measurement Processing Unpack/Pack Corruption The measurement processing software that interfaces with the receiver needs to unpack and repack the GNSS ephemeris. A software failure or network transmission failure could corrupt the GNSS ephemeris data and result in the SBAS using an incorrect ephemeris.

9.2.5 Estimation Errors

The SBAS system provides corrections to improve the accuracy of the GNSS measurements and to mitigate the GPS/GEO error sources. Estimations of parameters and corrections described in Sections 9.2.5.1–9.2.5.4 cause these errors, which must be accounted for.

9.2.5.1 Reference Time Offset Estimation Error The difference between the SBAS and GNSS reference time must be less than 50 ns. If the user is en route and mixing SBAS-corrected satellite data with non-SBAS-corrected satellite data, then the offset (error) between the SBAS reference time and the GNSS reference time could affect the user's receiver position solution. Under fault-free conditions, this error varies slowly. If one or more GNSS satellites fail, the offset between GNSS time and the SBAS reference time could vary rapidly.

9.2.5.2 Clock Estimation Error The SBAS system must compute estimates of the reference receiver clocks and GNSS/GEO satellite clock errors. An error in this estimation results in errors in the user's position solution. The error in the estimation process must be accounted for in the integrity bounds.

9.2.5.3 Ephemeris Correction Error The SBAS computes estimates of each satellite's orbit (ephemeris) and then uses these estimates to compute corrections. Error in the orbit (ephemeris) estimation process will result in erroneous corrections. Sources of error include measurement noise, troposphere modeling error, and orbital parameter modeling error. The error in the estimation process must be accounted for in the integrity bounds.

9.2.5.4 L1L2 Wide-Area Reference Equipment (WRE) and GPS Satellite Bias Estimation Error The L1L2 bias of the satellites and the receivers is used to generate the SBAS corrections. SBAS users utilize single-frequency corrections while corrections are generated using dual-frequency measurements that are unaffected by ionospheric delay errors. An error in the estimation process will result in erroneous corrections. Sources of estimation error include measurement error, time in view, ionospheric storms, and receiver/satellite malfunctions. The error in the estimation process must be accounted for in the integrity bounds.

9.2.6 Integrity-Bound Associated Errors

The integrity monitoring functionality in an SBAS system ensures that the system meets the allocated integrity requirement. This processing includes functionality that must be performed on a "trusted" platform with software developed to the proper RTCA/D0178-B safety level.

The ICAO HMI hazard has been evaluated to be a "hazardous/severe–major" failure condition. This requires all software responsible for preventing HMI to be developed using a process that meets all the RTCA/DO-178B Level B objectives.

A critical aspect of mitigating an integrity threat is the determination of the threat model. Threats originating in the RTCA/DO-178B Level B software can be characterized using observed performance, provided all the inputs originate from Level B software and the algorithms have been designed in an analytic methodology.

9.2.6.1 Ionospheric Modeling Errors The SBAS system uses an underlying characterization to transmit ionospheric corrections to the user. During periods of high solar activity, the ionospheric decorrelation can be quite rapid and large, and the true delay variation around the grid point may not match the underlying characterization. In this case, the SBAS-estimated delay measurement and the associated error bound may not be accurate or the SBAS may not sample a particular ionospheric event that is affecting a user.

9.2.6.2 Fringe Area Ephemeris Error Errors may be present in the SBAS GNSS position estimates that are not observable from the reference receivers. These errors could cause position errors in a user's position solution that are not observable to the reference receivers.

9.2.6.3 Small-Sigma Errors It is possible that any quantity of satellites could contain small- or medium-sized errors that combine in such a manner that creates an overall position error that is unbounded to a user.

9.2.6.4 Missed Message: Old But Active Data (OBAD) The user could have missed one or more messages and is allowed to use old corrections and integrity data. The use of these old data could result in an increased error compared to users that have not missed messages.

9.2.6.5 Time to Alarm (TTA) Exceeded If there is an underbound condition, the SBAS is required to correct that condition within a specified period of time. This is called the TTA. This alarm is a series of messages that contain the new information, such as an increased error bound or new corrections that are needed to correct the situation and prevent HMI. Different types of failure, such as hardware, software, or network transmission delay, could occur and cause the alarm messages to be delayed in excess of the required time.

9.2.7 GEO Uplink Errors

Errors caused by the uplink system can also be a source of HMI to the user.

9.2.7.1 GEO Uplink System Fails to Receive SBAS Message Any hardware or software along the path to the satellite could fault, causing the message to be delayed or not broadcast at all.

9.2.8 Mitigation of Integrity Threats

This section describes some approaches that may be used to eliminate and minimize data errors, mitigate integrity threats, and satisfy the safety integrity requirements.

Safety design and safety analyses are utilized to protect the data transmission path into the integrity monitors and out to the user through the geostationary satellite.

Such integrity monitors, written to DO-178B Level B standards to provide adjustments to the integrity bounds, must test the associated integrity data, user differential range error (UDRE) or grid ionosphere vertical error (GIVE) in an analytically tractable manner. The test prevents HMI by either passing the integrity data with no changes, increasing the integrity data to bound the residual error in the corrections, or setting the integrity data to "not monitored" or "don't use." Each integrity monitor must carefully account for the

uncertainty in each component of a calculation. Noisy measurements or poor quality corrections will result in large integrity bounds.

The examples given are for a system that utilizes either a "calculate then monitor" or "monitor then calculate" design. Both techniques are used in the examples to fully illustrate the types of mitigation needed to meet the general SBAS integrity requirements. Under the "calculate then monitor" design, corrections and error bounds are computed assuming that the inputs to the system follow some observed or otherwise predetermined model. A monitoring system then verifies the validity of these corrections and error bounds against the integrity threats. With the "monitor then calculate design," the measurements inputs to the monitor are carefully screened and forced to meet strict integrity requirements. The corrections and the error bounds are then computed in an analytically tractable manner and no further testing is required. Both designs must address all of the errors associated with an SBAS system in an analytically tractable manner.

9.2.8.1 Mitigation of GNSS Associated Errors

GNSS Clock Error

FAULT-FREE CASE The clock corrections are computed in a Kalman filter. The broadcast UDRE should be constructed using standard statistical techniques to ensure that the nominal errors in the fast corrections and long-term clock corrections are bounded.

FAULTED CASE A monitor is designed to ensure that the probability of a large fast correction error and/or long-term clock correction error is less than the allocation on the fault tree. The monitor must use measurements that are independent of the measurements used to compute the corrections. Error models for each input into the monitor must be determined and validated. The monitor either passes the UDRE or increases the UDRE or sets it to "not monitored" or "don't use" depending on the size of the GNSS clock error.

GPS Ephemeris Error

FAULT-FREE CASE The orbit corrections are computed in a Kalman filter. The broadcast UDRE would be constructed using standard statistical techniques to ensure that the nominal errors in the long-term position corrections are bounded.

FAULTED CASE Clock errors are easily observed by a differential GNSS system. The ability of an SBAS to observe orbit errors is dependent on the location of the system's reference stations. The SBAS can generate a covariance matrix and package it in SBAS message type 28. This message provides a location-specific multiplier for the broadcast UDRE. The covariance matrix must take into account the quality of the measurements from the reference stations and the quality of the ephemeris corrections broadcast from the SBAS. When the

GNSS ephemeris is grossly in error, the SBAS must either detect and correct the problem or increase the uncertainty in the UDRE. Under faulted conditions, the SBAS must account for the situation where clock error cancels with the ephemeris error at one or more of the reference stations.

GNSS Code and Carrier

FAULT-FREE CASE GNSS code-carrier divergence results from a failure on the GNSS satellite and errors do not need to be mitigated in the fault-free case.

FAULTED CASE A monitor must be developed to detect and alarm if the GNSS code and carrier phase become incoherent. The monitor must account for differences in the SBAS measurement smoothing algorithm and the user's measurement smoothing algorithm. The most difficult threat to detect and mitigate is one where the code-carrier divergence occurs shortly (within seconds) after the user acquires the satellite. In this case, the error has an immediate effect on the user and a gradual effect on the SBAS.

GNSS Signal Distortion

FAULT-FREE CASE GNSS signal distortion results from a failure on the GNSS satellite and errors do not need to be mitigated in the fault-free case.

FAULTED CASE A monitor can be developed to mitigate the errors from GNSS signal distortion. The measurement error incurred from signal distortion is receiver dependent. The monitor must mitigate the errors regardless of the type of equipment the user is employing.

GNSS L1L2 Bias

FAULT-FREE CASE L1L2 bias errors can be computed with a Kalman filter. These corrections are not sent to the user but are used in the other monitors. Nominal error bounds are computed with standard statistical techniques.

FAULTED CASE If the SBAS design utilizes the L1L2 bias corrections in the integrity monitors, then they must account for the faulted case. The L1L2 bias can suddenly change due to an equipment failure on board the GNSS satellite. The SBAS must be designed so that this type of failure does not "blind" the monitors. One approach to this design is to form a single-frequency integrity monitor that tests the corrections without using the L1L2 bias corrections.

Environment (Ionosphere) Errors

FAULT-FREE CASE Under calm ionospheric conditions, the GIVE is computed in a fashion that accounts for measurement uncertainty, L1L2 bias errors, and nominal fluctuations in the ionosphere.

FAULTED CASE The integrity monitors must ensure that an ionospheric storm cannot cause HMI. One approach to this problem is to create an ionospheric

storm detector that is sensitive to spatial and/or temporal changes in the ionospheric delay. Proving such a detector mitigates HMI is a difficult endeavor since the ionosphere is unpredictable during ionospheric storms. It is possible for ionospheric storms to exist in regions where the SBAS does not sample the event. An additional factor can be added to the GIVE to account for unobservable ionospheric storms. In some cases (when a reference receiver is out or the grid point is on the edge of the service volume), this term can be quite large. The GIVE must also account for rapid fluctuations in the ionosphere between ionospheric correction updates. One way to mitigate such errors is to run the monitor frequently and to send alarm messages if such an event occurs.

Environment (Troposphere) Errors

BOTH CASES Tropospheric delay errors are built into many of the SBAS corrections. The SBAS must determine error bounds on the tropospheric delay error and build them into the UDRE.

9.2.8.2 Mitigation of GEO-Associated Errors

GEO Code and Carrier and Environment Errors For GEO code-associated errors, fault-free and faulted, see Section 9.2.8.1, Subsection "GNSS Code and Carrier."

FAULT-FREE CASE Since GEO measurements are single frequency, the dual-frequency techniques utilized for GNSS integrity monitoring have to be modified. One approach to working with single-frequency measurements is to compensate for the iono delay using the broadcast ionospheric grid delays. The uncertainty of the iono corrections (GIVE) needs to be accounted for in the integrity monitors.

FAULTED CASE During ionospheric storms, the GIVE is likely to be substantially inflated. The inflated values will "blind" the other integrity monitors from detecting small GEO clock and ephemeris errors, resulting in a large GEO UDRE.

For both faulted and fault-free cases, of environment (troposphere) errors, see Section 9.2.8.1, Subsection "Environment (Troposphere) Errors, Both cases."

9.2.8.3 Mitigation of Receiver and Measurement Processing Errors

Receiver Measurement Error

FAULT-FREE CASE The integrity monitors must account for the noise in the reference station measurements. A bound on the noise can be computed and utilized in the integrity monitors. In the "calculate then monitor" approach, integrity monitors must use measurements that are uncorrelated with the measurements used to compute the corrections. Otherwise, error cancellation may occur.

FAULTED CASE In the faulted case, one or more receivers may be sending out erroneous measurements. An integrity monitor must be built to detect such events and to ensure that erroneous measurements do not blind the integrity monitors.

INTERCARD BIAS BOTH CASES Intercard bias errors appear to be measurement errors and are mitigated by the methods discussed in Section 9.2.4.1.

Code Noise and Multipath (CNMP)
FAULT-FREE CASE Small multipath errors are accounted for in the receiver measurement error discussed in Section 9.2.8.3.

FAULTED CASE Large multipath errors must be detected and screened from the integrity monitors or accounted for in the measurement noise error bounds.

WRE L1L2 Bias
FAULT-FREE CASE The WRE L1L2 bias can be computed in a manner similar to that for the GNSS L1L2 bias. The nominal errors in this computation must be bounded and accounted for in the integrity monitors.

FAULTED CASE A receiver can malfunction, causing the L1L2 bias to suddenly change. The L1L2 bias is used in the correction and integrity monitoring functions and such a change must be detected and corrected to prevent HMI. A single-frequency monitor can be created that tests the corrections without using L1L2 bias as an input.

WRE Clock Error
FAULT-FREE CASE The receiver clock error can be computed using a Kalman filter. Standard statistical techniques can be used to determine the error in the WRE clock estimates. This error bound can be utilized by the integrity monitors.

FAULTED CASE If bad data are received in the Kalman filter, erroneous WRE clock corrections could result. An integrity monitor can be built that does not utilize the WRE clock estimates from the Kalman filter to test the corrections when the WRE clock estimates are bad.

9.2.8.4 Mitigation of Estimation Errors

Reference Time Offset Estimation Error
FAULT-FREE CASE In the fault-free case, the difference between the GPS reference time and the SBAS reference time is accounted for by the user, provided the difference is less than 50 ns.

FAULTED CASE In the faulted case, due to some system fault or GPS anomaly, the difference in the SBAS reference time and the GPS reference time exceeds 50 ns. A simple monitor can be constructed to measure the difference between

the two references. The monitor would respond to a large offset by setting all satellites not monitored, stopping the user from mixing corrected and uncorrected satellites.

Clock Estimation Error, Ephemeris Correction Error, L1L2 WRE, and GNSS Satellite Bias Estimation Error
See Section 9.2.8.1, "GNSS Clock Error," "GNSS Ephemeris Error," and "GNSS L1L2 Bias," and Section 9.2.8.3, "WRE L1L2 Bias."

9.2.8.5 Mitigation of Integrity-Bound-Associated Errors

Ionospheric Modeling Error
FAULT-FREE CASE Extensive testing of the models used in the SBAS will provide assurance that the iono model error is properly bounding under quiet ionospheric conditions.

FAULTED CASE During an ionospheric storm, the validity of the model is in question. A monitor can be constructed to test the validity of the model and to increase the GIVE when the model is in question.

Fringe Area Ephemeris Error
FAULT-FREE CASE This error is mitigated by message type 28 as discussed in Section 9.2.8.1, "GNSS Ephemeris Error."

FAULTED CASE Special considerations must be taken to ensure that the integrity monitors are sensitive to satellite ephemeris errors on the fringe of coverage. Errors in the satellite ephemeris are not well viewed by the SBAS on the edge of the service region. A specific proof of the monitors' sensitivity to errors of this nature is required. Additional inflation factors may be needed to adjust the UDRE for this error.

Small-Sigma Errors
FAULT-FREE CASE Tests can easily be performed on individual corrections; the user, however, must be protected from the combination of all error sources. An analysis can be performed to demonstrate that any combination of errors observed in the fault-free case is bounded by the broadcast integrity bounds. An example of this analysis is discussed in Ref. 15.

FAULTED CASE Under faulted conditions, small biases may occur, which can "add" in the user position solution to cause HMI. This threat can be mitigated by monitoring the accuracy of the user position solution at the reference stations.

Missed Message: OBAD
FAULT-FREE CASE The OBAD deprivation factors broadcast by the SBAS account for aging data.

FAULTED CASE The integrity monitors must ensure that every combination of active SBAS messages meets the integrity requirements. Two methods are suggested for this threat. First, the integrity monitors can run on every active set of broadcast messages to check their validity after broadcast. If a large error is detected, an alarm will be sent. A second, preferable, approach is to test the messages against every active data set before broadcast and to adjust the corrections/integrity bounds accordingly.

TTA Exceeded

FAULT-FREE CASE The system is designed to meet the time-to-alarm requirement by continually monitoring the satellite signals and responding to integrity faults with alarms.

FAULTED CASE A monitor can be designed to test the "loop back" time in the system and continually ensure that the TTA requirement is met. The monitor sends a test message every minute and measures the time it takes for the message to loop back through the system.

9.3 SBAS EXAMPLE

The process for identifying, characterizing, and mitigating a failure condition is illustrated by the following SBAS example. SBAS broadcasts corrections to compensate for range errors incurred as the signal passes through the ionosphere. The uncertainty in these corrections is computed and sent to the user along with the corrections. HMI would result if the SBAS broadcasts erroneous integrity data (error bounds) and does not alert the user to the erroneous integrity data within a specified time limit. This time limit is referred to as the TTA.

1. *Identify Error Conditions that Can Cause HMI.* Error conditions can be caused by internal or external hardware or software failures or fluctuations in environmental conditions. The onset of an ionospheric storm represents a failure condition that could result in large errors in the ionospheric corrections, ultimately resulting in an increased probability of HMI.
2. *Precisely Characterize the Threat.* On days with nominal ionospheric behavior, the ionospheric threats are well understood and are reasonably easy to quantify. Scientists are not yet able to characterize the ionosphere during storm conditions. For these reasons, SBAS has generated specific threat models for the ionosphere based on real data collected during the worst ionospheric activity from the solar maximum period (an 11-year solar cycle). An important aspect of this model is the ionospheric irregularity detector, which assures the validity of the model and inflates the error bounds if the validity of the model is in question.

3. *Identify Error Detection Mechanisms.* In the SBAS, errors in ionospheric corrections are mitigated by a monitor located in a "safety processor" and a special detector called the "ionospheric irregularity detector."
4. *Analytically Determine That the Threat is Mitigated.* It is tempting to take an reliability, maintainability, availability (RMA) approach to dealing with ionospheric storms:
   (a) Ionospheric storms are "infrequent events."
   (b) "We haven't seen them cause HMI yet . . . ."
   (c) "They don't last very long."
   (d) "The system has other margins . . . ."

The a priori probability of a storm is not the mitigation of the threat. SBAS must meet its 10^{-7} integrity allocation during ionospheric storms. The analysis must account for worst-case events, like storms that are not well sampled by the ground system. Furthermore, it is not necessarily the storms with the highest magnitude that are the hardest to detect or are most likely to cause HMI. Extensive analysis is needed to characterize the threat.

In general, every requirement in a system's specification is tested by some type of formal demonstration. Most of the SBAS system-level requirements fall into this category; however, the SBAS integrity requirement does not. Testing fault-tree allocations of 10^{-7} and smaller requires on the order of 100,000,000 independent points (I sample every 5 min for 950 years). Integrity can only be demonstrated where reference stations exist.

Integrity must be proved for every satellite/user geometry. Every user at every point in space must be protected at all times. Demonstrations cannot be conducted where data are not available. In addition, every satellite geometry (subset) must be tested. Since GPS orbits repeat, then, if at a specific airport a satellite/user geometry exists with an increased probability of HMI, the situation will repeat every day at the same time until the constellation changes. It is because of these considerations that analytic proofs are required to satisfy integrity requirements.

The identification, characterization, and mitigation of a threat to the SBAS user should be carefully scrutinized by a panel of experts in the SBAS field. The analysis supporting claims is formally documented, scrutinized, and approved by this panel. This four-step process should be completed for every error identified in the system.

9.4 SUMMARY

The data used by an SBAS to calculate the corrections and integrity data are assumed to contain errors that have been sufficiently mitigated. The errors discussed are inherent in any SBAS design that utilizes GPS satellites. An SBAS design mitigates the majority of these errors with "corrections," thereby

making it a trusted navigation aid. The leftover errors, referred to as residual errors, are mitigated by the transmission of residual error bounding information. The threat of potential underbounding of integrity information is mitigated by integrity monitors. Both faulted and unfaulted cases are examined and mitigation strategies have been discussed. These SBAS corrections improve the accuracy of satellite signals. The integrity data ensure that the residual errors are bounded. The SBAS integrity monitors ensure that the integrity data have not been corrupted by SBAS failures. Following the integrity design guidelines given in this chapter is an important factor in obtaining certification and approval for use of the SBAS system.

SBAS integrity concepts may be applied to GBAS. In GBAS, the integrity will be broadcast from the ground.

9.5 FUTURE: GIC

A GNSS data integrity channel (GIC) will be provided in the next generation of GPS satellites such as GPS IIF and GPS III. In addition, the next generation will include airborne monitoring by using redundant measurements (RAIM). The GIC consists of a network of ground-based GNSS signal monitoring stations, located at known reference stations that cover a wide geographical area over which signal integrity is guaranteed by navigation providers, such as the FAA.

These monitors will be connected to a central control station where the integrity decision will be made. The integrity message will be broadcast through GEO stationary satellites [14].

PROBLEM

9.1 Use the data from GPS_Position(PRN#) from Appendix A for six satellites (position), pseudoranges, and user position to find the integrity using three RAIM methods.

REFERENCES

[1] R. B. Langley, "The Integrity of GPS," *GPS World*, 1999.

[2] B. W. Parkinson and P. Axelrad, "Autonomous GPS Integrity Monitoring Using the Pseudorange Residual," *Navigation* (Institute of Navigation) **35**(2), 255–274 (1988).

[3] B. W. Parkinson and J. J. Spilker, Jr. (Eds.), *Global Positioning System: Theory and Applications*, Vol. 1, Progress in Astronautics and Aeronautics (series). American Institute of Aeronautics and Astronautics, Washington, DC, 1996.

[4] B. W. Parkinson and J. J. Spilker, Jr. (Eds.), *Global Positioning System: Theory and Applications*, Vol. 2, Progress in Astronautics and Aeronautics (series). American Institute of Aeronautics and Astronautics, Washington, DC, 1996.

[5] B. W. Parkinson, M. L. O'Connor, and K. T. Fitzgibbon, "Aircraft Automatic Approach and Landing Using GPS," Chapter 14 in B. W. Parkinson and J. J. Spilker (Eds.), *Global Positioning System: Theory & Applications*, Vol. II, Progress in Astronautics and Aeronautics (series), Vol. 164, Paul Zarchan editor-in-chief. American Institute of Aeronautics and Astronautics, Washington, DC, 1995, pp. 397–425.

[6] Y. C. Lee, "Analysis of Range and Position Comparison Methods as a Means to Provide GPS Integrity in the User Receiver," *Proceedings of the Annual Meeting of the Institute of Navigation*, (Seattle, WA), June 24–26, 1986, pp. 1–4.

[7] RTCA, *Minimum Operational Performance Standards (MOPS) for Global Positioning System/Wide Area Augmentation System Airborne Equipment*, RTCA/DO-229, January 16, 1996, and subsequent changes, Appendix A, "WAAS System Signal Specification," RTCA, Washington, DC.

[8] M. A. Sturza, "Navigation System Integrity Monitoring Using Redundant Measurements," *Navigation* **35**(4), 483–501 (1988/89).

[9] M. A. Sturza and A. K. Brown, "Comparison of Fixed and Variable Threshold RAIM Algorithms," *Proceedings of the 3rd International Technical Meeting of the Institute of Navigation*, Satellite Division, ION GPS–90 (Colorado Springs, CO), September 19–21, 1990, pp. 437–443.

[10] J. E. Angus, "RAIM with Multiple Faults," *Journal of the Institute of Navigation* **53**(4), 249–257 (2006).

[11] C. A. Shively, et al., "Detailed Analysis of RAIM Performance for ADP-B Separation Error," *Journal of the Institute of Navigation* **56**(4), 261–274 (2009).

[12] C. Larson, et al., "The Impact of Altitude on Image Based Integrity," *Journal of the Institute of Navigation* **57**(4), 249–262 (2010).

[13] T. Schempp, et al., "WAAS Algorithm Contribution to Hazardously Misleading Information (HMI)," *14th International Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, Utah, September 11–14*, 2001.

[14] G. Watt, et al., "Lessons Learned in the Certification of Integrity for a Satellite-Based Navigation System," *Proceedings of the ION NTM 2003*, Anaheim, CA, January 2003.

[15] M. S. Grewal and A. P. Andrews, *Application of Kalman Filtering to GPS, INS, & Navigation*, Short Course Notes, Kalman Filtering Consulting Associates, Anaheim, CA, January 2013.

[16] T. Schempp, et al., "An Application of Gaussian Overbounding for the WAAS Fault-Free Error Analysis," *Proceedings of ION GPS*, Portland, OR, September 2002, pp. 766–772.

10

KALMAN FILTERING

Once you get the physics right, the rest is mathematics.[1]
—Rudolf E. Kalman

10.1 INTRODUCTION

The purpose of this chapter is to familiarize you with both the theoretical and practical aspects of Kalman filtering, especially those features that are necessary for global navigation satellite system (GNSS) navigation, and for GNSS/inertial navigation system (INS) integration. The presentation is primarily slanted toward these applications, but we have also included a brief derivation of the Kalman gain matrix, based on the maximum-likelihood estimation (MLE) model. We also include the derivation and demonstration of a method for assessing the relative statistical significance of linearization errors in GNSS implementations of the Kalman filter.

Broader treatments of the Kalman filter are presented in Refs. 1, 6, 10, 15, and 27; more basic introductions can be found in Refs. 8, 31; more mathematically rigorous derivations can be found in Ref. 14; more extensive coverage of the practical aspects of Kalman filtering can be found in Refs. 5 and 12; and background on matrices and matrix methods can be found in Refs. 2 and 11.

[1]Kailath Lecture, Stanford University, May 11, 2008.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

10.1.1 What Is a Kalman Filter?

It has been justifiably labeled "navigation's integration workhorse" [20] because it has become an essential part of modern navigation systems—especially for integrating navigation systems as disparate as GNSS and INS. It is unlikely that GNSS could have been developed without the Kalman filter, and certainly not GNSS/INS integration.

Earlier navigation implementations typically required the solution of sets of fairly complex equations giving the measured values of observable variables (e.g., sextant angles) as functions of the position variables in what is called a "forward model":

$$\text{measurement} = f(\text{position}).$$

Before Kalman filtering, solving for the independent variables (the navigation solution) as functions of the dependent variables (the measured values) required inverting that forward model.

The Kalman filter, on the other hand, uses the forward model directly. It does not require the functional inversion of the forward model. Its solution is a numerical algorithm using the equations of the forward model and their partial derivatives.

Kalman's paper introducing his now-famous filter was first published in 1960 [16], and its first practical implementation was for integrating an inertial navigator with airborne radar aboard the C5A military aircraft [21].

The application of interest here is quite similar. We want to integrate an onboard inertial navigator with a different electromagnetic ranging system (GNSS). There are many ways to do this [4], but nearly all involve Kalman filtering.

The Kalman filter is an extremely effective and versatile procedure for combining *noisy sensor outputs* to estimate the *state* of a *system* with *uncertain dynamics*, where

- The **noisy sensors** could be just *GNSS receivers* and *INSs*, but may also include subsystem-level sensors (e.g., GNSS clocks or INS accelerometers and gyroscopes) or auxiliary sensors such as speed sensors (e.g., wheel speed sensors for land vehicles, water speed sensors for ships, air speed sensors for aircraft, or Doppler radar), magnetic compasses, altimeters (barometric or radar) or radionavigation aids (e.g., distance measurement equipment [DME], VHF omnirange [VOR], long-range navigation [LORAN]).
- The **system state** in question may include the *position*, *velocity*, *acceleration*, *attitude*, and *attitude rate* of a *vehicle* on land, at sea, in the air, or in space. The system state may also include ancillary "nuisance variables" for modeling *time-correlated noise sources* such as ionospheric propagation delays of GNSS signals, and *time-varying parameters* of the sensors, GNSS receiver clock frequency and phase, or scale factors and output biases of accelerometers or gyroscopes.

Uncertain dynamics includes unpredictable disturbances of the host vehicle, whether caused by a human operator or by the medium (e.g., winds, surface currents, turns in the road, or terrain changes), but it may also include unpredictable changes in the sensor parameters.

10.1.2 How Does It Work?

The Kalman filter maintains two types of variables:

1. *An Estimate $\hat{\boldsymbol{x}}$ of the State Vector $\boldsymbol{x}$.* The components of the estimated state vector include the following:
   (a) The variables of interest (i.e., what we want or need to know, such as position and velocity).
   (b) Nuisance variables that are of no intrinsic interest but may be necessary to the estimation process. These nuisance variables may include, for example, the effective propagation delay errors in GNSS signals. We generally do not wish to know their values but may be obliged to calculate them to improve the receiver estimate of position.

   The Kalman filter state variables for a specific application ordinarily include all those system dynamic variables that are measurable by the sensors used in that application.

   For example, the Kalman filter state variables for GNSS-only navigation might include the ionospheric propagation delay errors in th pseudoranges between the satellites and the receiver antenna, or they might contain the position coordinates of the receiver antenna. Position could be represented by geodetic latitude, longitude, and altitude with respect to a reference ellipsoid, or geocentric latitude, longitude, and altitude with respect to a reference sphere, or earth-centered, earth-fixed (ECEF) Cartesian coordinates, or ECI coordinates, or any equivalent coordinates.

   In similar fashion, a Kalman filter for an INS containing accelerometers and rate gyroscopes might include accelerations and rotation rates to which these instruments respond. However, simplified INS models might ignore the accelerometers and angular rate sensors and model the INS itself as a position-only sensor, or as a position and velocity sensor.
2. *An Estimate of Estimation Uncertainty.* Uncertainty is modeled by a *covariance matrix*

$$\mathbf{P} \overset{\text{def}}{=} E\left\langle (\hat{\mathbf{x}} - \mathbf{x})(\hat{\mathbf{x}} - \mathbf{x})^{\mathrm{T}} \right\rangle \tag{10.1}$$

   of estimation error $(\hat{\mathbf{x}} - \mathbf{x})$, where $\hat{\mathbf{x}}$ is the estimated state vector, $\mathbf{x}$ is the actual state vector, and E is the expectancy operator.

The equations used to propagate the covariance matrix (collectively called the **Riccati equation**) model and manage *uncertainty*, taking into account how sensor noise and dynamic uncertainty contribute to uncertainty about the estimated system state.

By maintaining an estimate of its own estimation uncertainty and the relative uncertainty in the various sensor outputs, the Kalman filter is able to combine all sensor information "optimally," in the sense that the resulting estimate minimizes any linear quadratic loss function of estimation error, including the mean-squared value of any linear combination of state estimation errors. The *Kalman gain* is the optimal weighting matrix for combining new sensor information with a prior estimate to obtain a new estimate. The Kalman gain is usually obtained as a partial result in the solution of the Riccati equation.

10.1.2.1 Prediction and Correction The Kalman filter is a two-step process, the steps of which we call "prediction" and "correction." The filter can start with either step.

The **correction step** updates the estimate itself and the covariance matrix of estimation uncertainty, based on new information obtained from sensor measurements. It is also called the *observational update* or *measurement update*, and the Latin prepositional phrase **a posteriori** is often used for the corrected estimate and its associated uncertainty.

The **prediction step** updates the estimate and estimation uncertainty, taking into account the effects of uncertain system dynamics over the times between measurements. It is also called the *temporal update*, and the Latin phrase **a priori** is often used for the predicted estimate and its associated uncertainty.

10.1.3 How Is It Used?

1. **For estimation**, using the full complement of variables:
   (a) The estimate $\hat{\mathbf{x}}$ of the state of a stochastic system, using noisy measurements $\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v}$, related to the state variables $\mathbf{x}$ of the stochastic system. The **stochastic system** is modeled as a **discrete-time Markov process**: $\mathbf{x}_k = \mathbf{\Phi}_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}$, where the matrices $\mathbf{\Phi}_{k-1}$ are known, but the "process noise" variables $\mathbf{w}_{k-1}$ are modeled as vector-valued zero-mean white-noise processes.
   (b) The successive estimates $\mathbf{P}_k$ of uncertainties in the estimates $\hat{\mathbf{x}}_k$ of the state vector. These values are needed for calculating the Kalman gains, which apply the statistically optimal weighting to the information in the measurements, based on the relative uncertainties in estimated state variables and measurements $\mathbf{z}_k$ (due to the noise $\mathbf{v}_k$).

   It has become the essential tool for implementing GNSS systems and for integrating GNSS with INSs.

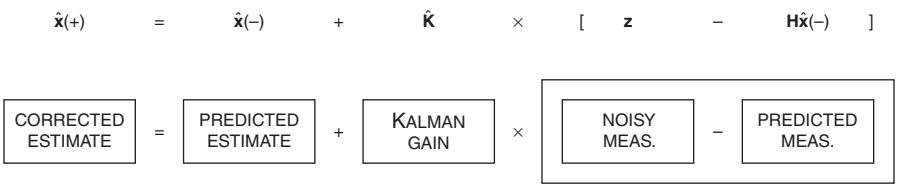


Fig. 10.1 Estimate correction using Kalman gain.

2. **For predicting performance of a proposed sensor system for a particular mission**, using only the Riccati equations for calculating the successive values of $\mathbf{P}_k$. It turns out that the performance of the Kalman filter, in terms of how well it can estimate the unknown state variables based on the measurements, does not depend on the measurements themselves. Performance depends only on values of the matrices $\mathbf{\Phi}_{k-1}$ and $\mathbf{H}_k$, plus the respective covariance matrices associated with the two white-noise processes $\mathbf{w}_k$ and $\mathbf{v}_k$. As a consequence, the expected performance of a proposed sensor system design for a specified stochastic system can be determined by solving only the Riccati equations ancillary to the Kalman filter. **This form of systems analysis has been of enormous value.** It was the essential tool used in the design of the GPS navigation system, for evaluating alternative satellite constellations and alternative hardware and software implementations in the satellites and receivers.

10.2 KALMAN FILTER CORRECTION UPDATE

The Kalman gain matrix $\bar{\mathbf{K}}$ is the crown jewel of Kalman filtering. All the effort of solving the matrix Riccati equation is for the purpose of computing the "optimal" value of the gain matrix $\bar{\mathbf{K}}$ used as shown in Fig. 10.1 for correcting an estimate $\hat{\mathbf{x}}$, based on the measurement

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \text{noise} \tag{10.2}$$

that is a linear function of the vector variable $\mathbf{x}$ to be estimated, plus additive noise with known statistical properties.

10.2.1 Deriving the Kalman Gain

Like laws and sausages,[2] people who like Kalman filtering often find its derivation somewhat traumatic. This led academicians to develop alternative derivations, more easily understood by engineering students. What is perhaps the

[2]A reference to the observation that people who like laws and sausages should not watch them being made, often attributed to Otto von Bismarck (1815–1898) but traced to the American poet John Saxe (1816–1887), author of "The Blind Men and the Elephant."

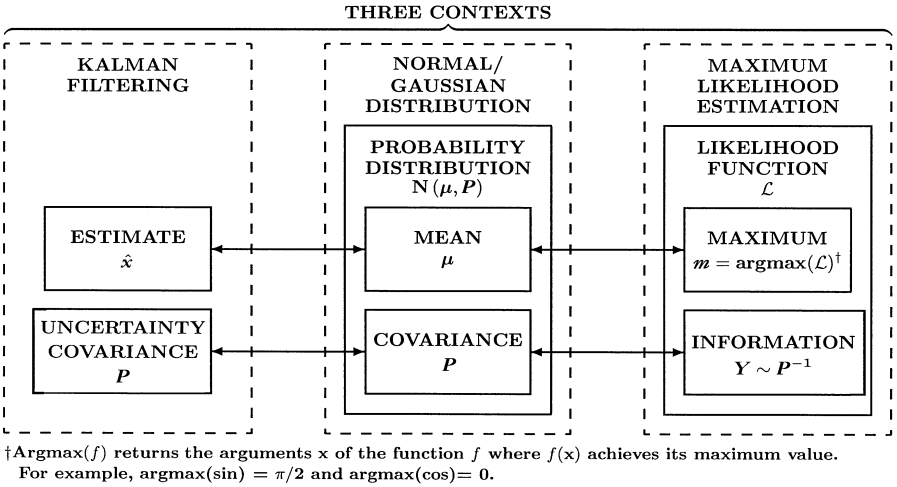


Fig. 10.2 Analogous concepts in three different contexts.

most user-friendly of these, using the maximum-likelihood estimator model, is given below.

10.2.1.1 Approaches to Deriving the Kalman Gain Kalman's original derivation of his gain matrix made no assumptions about the underlying probability distributions, but this requires a level of mathematical rigor that is a bit beyond standard engineering mathematics. As an alternative, it has become common practice to derive the formula for the Kalman gain matrix $\overline{\mathbf{K}}$ based on an analogous filter called the **linear Gaussian quadratic maximum-likelihood estimator** (LGQ-MLE). It uses the analogies shown in Fig. 10.2 between concepts in Kalman filtering, Gaussian probability distributions, and MLE.

This derivation is given in the following subsections. It begins with some background information on the properties of Gaussian probability distributions and Gaussian likelihood functions, then development of models for noisy sensor outputs and a derivation of the associated **maximum-likelihood estimate** (MLE) for combining prior estimates with noisy sensor measurements.

10.2.1.2 Gaussian Probability Density Functions **Probability density functions** (PDFs) are nonnegative integrable functions whose integral equals *unity* (i.e., 1). The density functions of *Gaussian* probability distributions all have the form

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det \mathbf{P}}} \exp\left(-\frac{1}{2}[\mathbf{x}-\boldsymbol{\mu}]^{\mathrm{T}} \mathbf{P}^{-1}[\mathbf{x}-\boldsymbol{\mu}]\right), \tag{10.3}$$

where n is the dimension of P (i.e., P is an $n \times n$ matrix) and the parameters

$$\boldsymbol{\mu} \stackrel{\text{def}}{=} \mathop{\mathrm{E}}_{\mathbf{x} \in N(\boldsymbol{\mu}, P)} \langle \mathbf{x} \rangle \tag{10.4}$$

$$\stackrel{\text{def}}{=} \int_{x_1} dx_1 \cdots \int_{x_n} dx_n p(\mathbf{x}) \mathbf{x} \tag{10.5}$$

$$\mathbf{P} \stackrel{\text{def}}{=} \mathop{\mathrm{E}}_{x \in N(\boldsymbol{\mu}, P)} \left\langle (\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^{\mathrm{T}} \right\rangle \tag{10.6}$$

$$\stackrel{\text{def}}{=} \int_{x_1} dx_1 \cdots \int_{x_n} dx_n p(\mathbf{x})(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^{\mathrm{T}}. \tag{10.7}$$

The parameter $\boldsymbol{\mu}$ is the *mean* of the distribution. It will be a column vector with the same dimensions as the variate $\mathbf{x}$.

The parameter $\mathbf{P}$ is the *covariance matrix* of the distribution. By its definition, it will always be an $n \times n$, *symmetric*, *nonnegative definite* matrix. However, because its determinant appears in the denominator of the square root and its inverse appears in the exponential function argument, it must be *positive definite* as well; that is, its eigenvalues must be real and positive for the definition to work.

The constant factor $1/\sqrt{(2\pi)^n \det \mathbf{P}}$ in Eq. 10.3 is there to make the integral of the probability density function equal to unity, a necessary condition for all probability density functions.

The operator $E\langle \cdot \rangle$ is the **expectancy operator**, also called the **expected-value operator**.

The notation $\mathbf{x} \in N(\boldsymbol{\mu}, \mathbf{P})$ denotes that the *variate* (i.e., random variable) $\mathbf{x}$ is drawn from the Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\mathbf{P}$. Gaussian distributions are also called *normal* (the source of the N notation) or *Laplace* distributions.

10.2.1.3 Properties of Likelihood Functions Likelihood functions are similar to probability density functions except that their integrals are not constrained to equal unity or even required to be finite. They are useful for comparing *relative* likelihoods and for finding a value,

$$\mathbf{m} \in \operatorname{argmax}[\mathcal{L}(\mathbf{x})], \tag{10.8}$$

of the unknown independent variable $\mathbf{x}$ at which the likelihood function $\mathcal{L}$ achieves its maximum,[3] as illustrated in Fig. 10.3.

[3]It is possible that a likelihood function will achieve its maximum value at more than one value of $\mathbf{x}$, but that will not matter in the derivation.

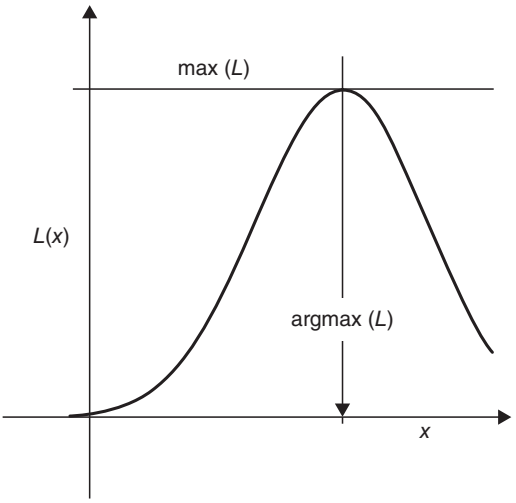


Fig. 10.3 Maximum-likelihood estimate.

Gaussian Likelihood Functions These functions have the form

$$\mathcal{L}(\mathbf{x}, \mathbf{m}, \mathbf{Y}) = c \exp\left\{-\frac{1}{2}[\mathbf{x}-\mathbf{m}]^{\mathrm{T}}\,\mathbf{Y}\,[\mathbf{x}-\mathbf{m}]\right\}, \tag{10.9}$$

where $c > 0$ is an arbitrary scaling variable and $\mathbf{m}$ (defined in Eq. 10.8) is a value of $\mathbf{x}$ at which $\mathcal{L}$ achieves its maximum value.

Information Matrices The parameter $\mathbf{Y}$ in Eq. 10.9 is called the *information matrix* of the likelihood function. It replaces $\mathbf{P}^{-1}$ in the Gaussian probability density function. If the information matrix $\mathbf{Y}$ is nonsingular, then its inverse $\mathbf{Y}^{-1} = \mathbf{P}$, a covariance matrix. However, **an information matrix is not required to be nonsingular**. This property of information matrices is important for representing the information from a set of measurements (sensor outputs) with incomplete information for determining the unknown vector $\mathbf{x}$; that is, the measurements may provide *no information* about some linear combinations of the components of $\mathbf{x}$, as illustrated in Fig. 10.4.

Scaling of Likelihood Functions MLE is based on the argmax of the likelihood function, but for any positive scalar $c > 0$,

$$\arg\max(c\mathcal{L}) = \arg\max(\mathcal{L}); \tag{10.10}$$

that is, positive scaling of likelihood functions will have no effect on the maximum-likelihood estimate. As a consequence, likelihood functions can have arbitrary positive scaling.

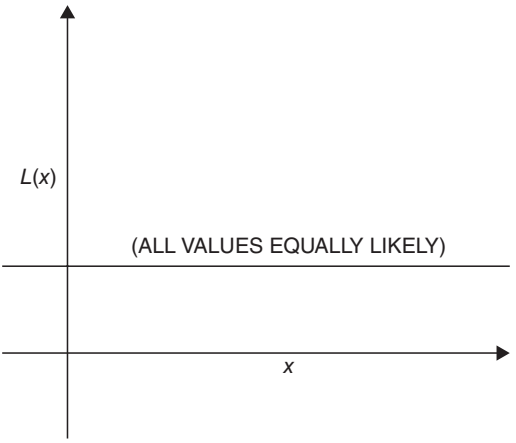


Fig. 10.4 Likelihood without unique maximum.

Independent Likelihood Functions The joint probability $P(A\&B)$ of independent events A and B is the product $P(A\&B) = P(A) \times P(B)$. The analogous effect on independent likelihood functions $\mathcal{L}_A$ and $\mathcal{L}_B$ is the pointwise product; that is, at each "point" $\mathbf{x}$,

$$\mathcal{L}_{A\&B}(\mathbf{x}) = \mathcal{L}_A(\mathbf{x}) \times \mathcal{L}_B(\mathbf{x}). \tag{10.11}$$

Pointwise Products of Likelihood Functions One of the remarkable attributes of Gaussian likelihood functions is that their pointwise products are also Gaussian likelihood functions, as illustrated in Fig. 10.5.

A Lemma Given two Gaussian likelihood functions with parameter sets $\{\mathbf{m}_A, \mathbf{Y}_A\}$ and $\{\mathbf{m}_B, \mathbf{Y}_B\}$, their pointwise product is a scaled Gaussian likelihood function with parameters $\{\mathbf{m}_{A\&B}, \mathbf{Y}_{A\&B}\}$ such that, for all $\mathbf{x}$,

$$\exp\left(-\frac{1}{2}[\mathbf{x}-\mathbf{m}_{A\&B}]^{\mathrm{T}}\mathbf{Y}_{A\&B}[\mathbf{x}-\mathbf{m}_{A\&B}]\right) = c \times \exp\left(-\frac{1}{2}[\mathbf{x}-\mathbf{m}_A]^{\mathrm{T}}\mathbf{Y}_A[\mathbf{x}-\mathbf{m}_A]\right) \times \exp\left(-\frac{1}{2}[\mathbf{x}-\mathbf{m}_B]^{\mathrm{T}}\mathbf{Y}_B[\mathbf{x}-\mathbf{m}_B]\right) \tag{10.12}$$

for some constant $c > 0$.

This is the fundamental lemma about Gaussian likelihood functions, and proving it by construction will give us the Kalman gain matrix as a corollary.

10.2.1.4 Solving for Combined Information Matrix One can solve Eq. 10.12 for the parameters $\mathbf{m}_{A\&B}$ and $\mathbf{Y}_{A\&B}$ as functions of the parameters $\mathbf{m}_A$, $\mathbf{Y}_A$, $\mathbf{m}_B$, and $\mathbf{Y}_B$.

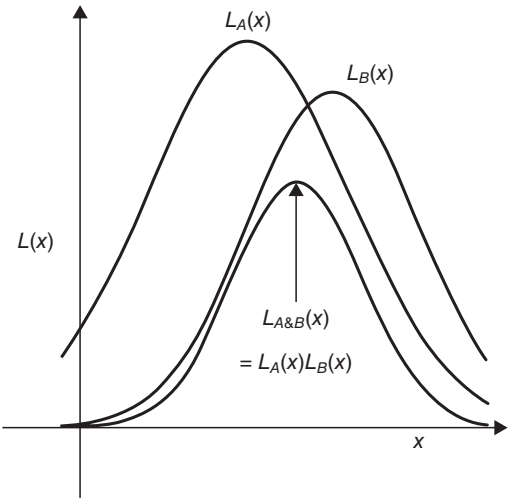


Fig. 10.5 Pointwise products of Gaussian likelihood functions.

Taking logarithms of both sides of Eq. 10.12 will produce the equation

$$-\frac{1}{2}[\mathbf{x}-\mathbf{m}_{A\&B}]^{\mathrm{T}}\mathbf{Y}_{A\&B}[\mathbf{x}-\mathbf{m}_{A\&B}] = \log(c) - \frac{1}{2}[\mathbf{x}-\mathbf{m}_A]^{\mathrm{T}}\mathbf{Y}_A[\mathbf{x}-\mathbf{m}_A] - \frac{1}{2}[\mathbf{x}-\mathbf{m}_B]^{\mathrm{T}}\mathbf{Y}_B[\mathbf{x}-\mathbf{m}_B]. \tag{10.13}$$

Next, taking the first and second derivatives with respect to the independent variable $\mathbf{x}$ will produce the equations

$$\mathbf{Y}_{A\&B}(\mathbf{x}-\mathbf{m}_{A\&B}) = \mathbf{Y}_A(\mathbf{x}-\mathbf{m}_A) + \mathbf{Y}_B(\mathbf{x}-\mathbf{m}_B), \tag{10.14}$$

$$\mathbf{Y}_{A\&B} = \mathbf{Y}_A + \mathbf{Y}_B, \tag{10.15}$$

respectively.

Information is Additive The information matrix of the combined likelihood function ($\mathbf{Y}_{A\&B}$ in Eq. 10.15) equals the sum of the individual information matrices of the component likelihood functions ($\mathbf{Y}_A$ and $\mathbf{Y}_B$ in Eq. 10.15).

10.2.1.5 Solving for Combined Argmax Equation 10.14 evaluated at $\mathbf{x} = 0$ is

$$\mathbf{Y}_{A\&B}\mathbf{m}_{A\&B} = \mathbf{Y}_A\mathbf{m}_A + \mathbf{Y}_B\mathbf{m}_B, \tag{10.16}$$

which can be solved for

$$\mathbf{m}_{A\&B} = \mathbf{Y}_{A\&B}^{\dagger}(\mathbf{Y}_A\mathbf{m}_A + \mathbf{Y}_B\mathbf{m}_B) \tag{10.17}$$

$$= (\mathbf{Y}_A + \mathbf{Y}_B)^{\dagger}(\mathbf{Y}_A\mathbf{m}_A + \mathbf{Y}_B\boldsymbol{\mu}_B), \tag{10.18}$$

where † denotes the Moore–Penrose pseudoinverse[4] of a matrix.

The Combined Maximum-Likelihood Estimate is a Weighted Average Equations 10.15 and 10.18 are key results for deriving the formula for Kalman gain. All that remains is to define likelihood function parameters for noisy sensors.

10.2.1.6 Noisy Measurement Likelihoods The term *measurement* refers to outputs of sensors that are to be used in estimating the argument vector **x** of a likelihood function. Measurement models represent how these measurements are related to **x**, including those errors called *measurement errors* or *sensor noise*. These models can be expressed in terms of likelihood functions with **x** as the argument.

The Measurement Vector The collective output values from a multitude ℓ of sensors can be represented as the components of a vector,

$$\mathbf{z} \overset{\text{def}}{=} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \vdots \\ z_\ell \end{bmatrix}, \tag{10.19}$$

called the *measurement vector*, a column vector with ℓ rows.

Measurement Sensitivity Matrix We suppose that the measured values z_i are linearly[5] related to the unknown vector **x** we wish to estimate,

$$\mathbf{z} = \mathbf{H}\mathbf{x}, \tag{10.20}$$

where **H** is the measurement sensitivity matrix.

[4]The Moore–Penrose pseudoinverse $\mathbf{M}^{\dagger}$ of a matrix **M** satisfies the four conditions:

$$\mathbf{M}\mathbf{M}^{\dagger}\mathbf{M} = \mathbf{M}, \mathbf{M}^{\dagger}\mathbf{M}\mathbf{M}^{\dagger} = \mathbf{M}^{\dagger}, (\mathbf{M}\mathbf{M}^{\dagger})^{\mathrm{T}} = \mathbf{M}\mathbf{M}^{\dagger}, (\mathbf{M}^{\dagger}\mathbf{M})^{\mathrm{T}} = \mathbf{M}^{\dagger}\mathbf{M},$$

where $^{\mathrm{T}}$ denotes the matrix transpose (or Hermitian transpose of complex matrices). The Moore–Penrose pseudoinverse is defined for all matrices, including nonsquare matrices and nonsingular matrices.

[5]The Kalman filter is defined in terms of the *measurement sensitivity matrix* **H**, but the *extended Kalman filter* can be defined in terms of a suitably differentiable vector-valued function **h**(**x**).

Measurement Noise Measurement noise is the unpredictable error at the output of the sensors. It is also called "sensor noise." It is assumed to be additive:

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v} \tag{10.21}$$

or

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{J}\mathbf{v}, \tag{10.22}$$

where the measurement noise vector $\mathbf{v}$ is assumed to be zero-mean Gaussian noise with known covariance $\mathbf{R}$:

$$E\langle \mathbf{v} \rangle \stackrel{\text{def}}{=} 0, \tag{10.23}$$

$$\mathbf{R} \stackrel{\text{def}}{=} E\langle \mathbf{v}\mathbf{v}^{\mathrm{T}} \rangle. \tag{10.24}$$

Sensor Noise Distribution Matrix The matrix $\mathbf{J}$ in Eq. 10.22 is called a "sensor noise distribution matrix." It models "common mode" sensor noise, in which a lower-dimensional noise source (e.g., power supply noise, electromagnetic interference, or temperature variations) corrupts multiple sensor outputs.

Measurement Likelihood A measurement vector $\mathbf{z}$ and its associated covariance matrix of measurement noise $\mathbf{R}$ define a likelihood function for the "true" value of the measurement (i.e., without noise). This likelihood function will have its argmax at

$$\mathbf{m}_z = \mathbf{z} \tag{10.25}$$

and information matrix

$$\mathbf{Y}_z = \mathbf{R}^{-1}, \tag{10.26}$$

assuming $\mathbf{R}$ is nonsingular.

Unknown Vector Likelihoods The same parameters defining measurement likelihoods also define an inferred likelihood function for the true value of the unknown vector, with argmax

$$\mathbf{m}_x = \mathbf{H}^{\dagger}\mathbf{m}_z \tag{10.27}$$

$$= \mathbf{H}^{\dagger}\mathbf{z} \tag{10.28}$$

and information matrix

$$\mathbf{Y}_x = \mathbf{H}^{\mathrm{T}}\mathbf{Y}_z\mathbf{H} \tag{10.29}$$

$$= \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}, \tag{10.30}$$

where the $n \times \ell$ matrix $\mathbf{H}^{\dagger}$ is defined as the Moore–Penrose pseudoinverse of the $\ell \times n$ matrix $\mathbf{H}$. This information matrix will be singular if $\ell < n$ (i.e., there are fewer sensor outputs than unknown variables), which is not unusual for GNSS/INS integration.

10.2.1.7 Gaussian Maximum-Likelihood Estimate (MLE)

Variables The Gaussian MLE uses the following variables:

- $\hat{\mathbf{x}}$, the maximum-likelihood estimate of $\mathbf{x}$. It will always equal the argmax of an associated Gaussian likelihood function, but it can have different values:
  - $\hat{\mathbf{x}}(-)$, the predicted value, representing the likelihood function prior to using the measurement results.
  - $\hat{\mathbf{x}}(+)$, the corrected value, representing the likelihood function after using the measurement results.
- $\mathbf{P}$, the covariance matrix of estimation uncertainty. It will always equal the inverse of the information matrix $\mathbf{Y}$ of the associated likelihood function. It also can have two values:
  - $\mathbf{P}(-)$, representing the likelihood function prior to using the measurements.
  - $\mathbf{P}(+)$, representing the likelihood function after using the measurements.
- $\mathbf{z}$, the vector of measurements.
- $\mathbf{H}$, the measurement sensitivity matrix.
- $\mathbf{R}$, the covariance matrix of sensor noise.

Maximum-Likelihood Correction Equations The MLE formula for correcting the variables $\hat{\mathbf{x}}$ and $\mathbf{P}$ to reflect the effect of measurements can be derived from Eqs. 10.15 and 10.18 with initial likelihood parameters

$$\mathbf{m}_A = \hat{\mathbf{x}}(-), \tag{10.31}$$

the MLE before measurements, and

$$\mathbf{Y}_A = \mathbf{P}(-)^{-1}, \tag{10.32}$$

the inverse of the covariance matrix of MLE uncertainty before measurements. The likelihood function of $\mathbf{x}$ inferred from the measurements alone (i.e.,

without taking into account the prior estimate) is represented by the likelihood function parameters

$$\mathbf{Y}_B = \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}, \tag{10.33}$$

the information matrix of the measurements, and

$$\mathbf{m}_B = \mathbf{H}^{\dagger}\mathbf{z}, \tag{10.34}$$

where $\mathbf{z}$ is the measurement vector and † represents the Moore–Penrose pseudoinverse.

Covariance Update The Gaussian likelihood function with parameters $\mathbf{m}_{A\&B}$, $\mathbf{Y}_{A\&B}$ of Eqs. 10.15 and 10.18 then represents the state of knowledge about the unknown vector $\mathbf{x}$ combining both sources (i.e., the prior likelihood and the effect of the measurements); that is, the covariance of MLE uncertainty after using the measurements will be

$$\mathbf{P}(+) = \mathbf{Y}_{A\&B}^{-1}, \tag{10.35}$$

and the MLE of $\mathbf{x}$ after using the measurements will then be

$$\hat{\mathbf{x}}(+) = \mathbf{m}_{A\&B}. \tag{10.36}$$

Equation 10.15 can be simplified by applying the following general matrix formula[6]:

$$(\mathbf{A}^{-1} + \mathbf{B}\mathbf{C}^{-1}\mathbf{D})^{-1} = \mathbf{A} - \mathbf{A}\mathbf{B}(\mathbf{C} + \mathbf{D}\mathbf{A}\mathbf{B})^{-1}\mathbf{D}\mathbf{A}, \tag{10.37}$$

where

$$\left.\begin{aligned} \mathbf{A}^{-1} &= \mathbf{Y}_A, \text{ the prior information matrix for } \hat{\mathbf{x}} \\ \mathbf{A} &= \mathbf{P}(-), \text{ the prior covariance matrix for } \hat{\mathbf{x}} \\ \mathbf{B} &= \mathbf{H}^{\mathrm{T}}, \text{ the transpose of the measurement sensitivity matrix} \\ \mathbf{C} &= \mathbf{R} \\ \mathbf{D} &= \mathbf{H}, \text{ the measurement sensitivity matrix,} \end{aligned}\right\} \tag{10.38}$$

so that Eq. 10.37 becomes

[6]A formula with many discoverers. Henderson and Searle [13] list some earlier ones.

$$\left.\begin{aligned}
\mathbf{P}(+) &= \mathbf{Y}_{A\&B}^{-1} \\
&= (\mathbf{Y}_A + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H})^{-1} && \text{(Eq. 10.15)} \\
&= \mathbf{Y}_A^{-1} - \mathbf{Y}_A^{-1}\mathbf{H}^{\mathrm{T}}(\mathbf{H}\mathbf{Y}_A^{-1}\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1}\mathbf{H}\mathbf{Y}_A^{-1} && \text{(Eq. 10.37)} \\
&= \mathbf{P}(-) - \mathbf{P}(-)\mathbf{H}^{\mathrm{T}}(\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1}\mathbf{H}\mathbf{P}(-),
\end{aligned}\right\} \tag{10.39}$$

a form better conditioned for computation.

10.2.1.8 Estimate Correction Equation 10.18 with substitutions from Eqs. 10.31–10.34 will have the form shown in Fig. 10.1:

$$\left.\begin{aligned}
\hat{\mathbf{x}}(+) &= \mathbf{m}_{A\&B} && \text{(Eq. 10.36)} \\
&= (\mathbf{Y}_A + \mathbf{Y}_B)^{\dagger}(\mathbf{Y}_A\mathbf{m}_A + \mathbf{Y}_B\mathbf{m}_B) && \text{(Eq. 10.18)} \\
&= \underbrace{\mathbf{P}(+)}_{(10.35)}[\underbrace{\mathbf{P}(-)^{-1}}_{(10.32)}\underbrace{\hat{\mathbf{x}}(-)}_{(10.31)} + \underbrace{\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}}_{(10.33)}\underbrace{\mathbf{H}^{\dagger}z}_{(10.34)}] \\
&= [\mathbf{P}(-) - \mathbf{P}(-)\mathbf{H}^{\mathrm{T}}(\mathbf{H}P(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1}\mathbf{H}P(-)] \\
&\quad \times [\mathbf{P}(-)^{-1}\hat{\mathbf{x}}(-) + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}\mathbf{H}^{\dagger}\mathbf{z}] && \text{(Eq. 10.39)} \\
&= [\mathbf{I} - \mathbf{P}(-)\mathbf{H}^{\mathrm{T}}(\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1}\mathbf{H}] \\
&\quad \times [\hat{\mathbf{x}}(-) + \mathbf{P}(-)\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}\mathbf{H}^{\dagger}\mathbf{z}] \\
&= \hat{\mathbf{x}}(-) + \mathbf{P}(-)\mathbf{H}^{\mathrm{T}}(\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1} \\
&\quad \times [(\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R})\mathbf{R}^{-1} - \mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}]\{\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}(-)\} \\
&= \hat{\mathbf{x}}(-) + \mathbf{P}(-)\mathbf{H}^{\mathrm{T}}(\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1} \\
&\quad \times [\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1} + \mathbf{I} - \mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}]\{\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}(-)\} \\
&= \hat{\mathbf{x}}(-) + \underbrace{\mathbf{P}(-)H^{\mathrm{T}}(\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R})^{-1}}_{\bar{\mathbf{K}}} \times \{\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}(-)\},
\end{aligned}\right\}, \tag{10.40}$$

where the matrix $\bar{\mathbf{K}}$ has a special meaning in Kalman filtering.

10.2.1.9 Kalman Gain Matrix for MLE The last line in Eq. 10.40 has the form of the equation in Fig. 10.1 with Kalman gain matrix

$$\bar{\mathbf{K}} = \mathbf{P}(-)\mathbf{H}^{\mathrm{T}}[\mathbf{H}\mathbf{P}(-)\mathbf{H}^{\mathrm{T}} + \mathbf{R}]^{-1}, \tag{10.41}$$

which completes the derivation of the Kalman gain matrix based on Gaussian MLE.

10.2.2 Estimate Correction Using the Kalman Gain

The Kalman gain expression from Eq. 10.41 can be substituted into Eq. 10.40 to yield

$$\hat{\mathbf{x}}(+) = \hat{\mathbf{x}}(-) + \bar{\mathbf{K}}[\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}(-)], \tag{10.42}$$

the estimate correction equation to account for the effects of measurements.

10.2.3 Covariance Correction for Using Measurements

The act of making a measurement and correcting the estimate based on the information obtained reduces the uncertainty about the estimate. The effect this has on the covariance of estimation uncertainty **P** can be found by substituting Eq. 10.41 into Eq. 10.39. The result is a simplified equation for the covariance matrix update to correct for the effects of using the measurements:

$$\mathbf{P}(+) = \mathbf{P}(-) - \bar{\mathbf{K}}\mathbf{H}\mathbf{P}(-). \tag{10.43}$$

10.3 KALMAN FILTER PREDICTION UPDATE

The rest of the Kalman filter is the prediction step, in which the estimate $\hat{\mathbf{x}}$ and its associated covariance matrix of estimation uncertainty **P** are propagated from one time epoch to another. This is the part where the dynamics of the underlying physical processes come into play. The "state" of a dynamic process is a vector of variables that completely specify enough of the initial boundary value conditions for propagating the trajectory of the dynamic process forward in time, and the procedure for propagating that solution forward in time is called "state prediction." The model for propagating the covariance matrix of estimation uncertainty is derived from the model used for propagating the state vector.

10.3.1 Stochastic Systems in Continuous Time

The word *stochastic* derives from the Greek expression for *aiming at a target*, indicating some degree of uncertainty in the dynamics of the projectile between launch and impact. That idea has been formalized mathematically as *stochastic systems theory*, in which a *stochastic process* is a model for the evolution over time of a *probability distribution*.

10.3.1.1 White-Noise Processes A white-noise process in continuous time is a function whose value at any time is a sample from a zero-mean Gaussian distribution, statistically independent of the values sampled at other times. White-noise processes are not integrable functions in the ordinary (Riemann) calculus. A special calculus is required to render them integrable. It is called the *stochastic calculus*. See Ref. 14 for more on this.

10.3.1.2 Stochastic Differential Equations Ever since the differential calculus was introduced (more or less contemporaneously) by Sir Isaac Newton (1643–1727) and Gottfried Wilhelm Leibnitz (1646–1716), we have been using ordinary differential equations as models for the dynamical behavior of systems of all sorts.

In 1827, botanist Robert Brown (1773–1858) described the apparently random motions of very small particles immersed in fluids, and the phenomenon came to be called *Brownian motion*. In 1908, French physicist Paul Langevin[7] (1872–1946) published a mathematical model for Brownian motion as a differential equation [19]. It is now called the *Langevin equation*. It includes a random function of time that was eventually characterized as a *white-noise process*. When the dependent variables in a differential equation include white-noise processes $\mathbf{w}(t)$, it is the first known example of what is now called a *stochastic differential equation*.

Uncertain dynamical systems are modeled by linear stochastic differential equations of the sort

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{w}(t) \tag{10.44}$$

or

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t), \tag{10.45}$$

where

- $\mathbf{x}(t)$ is the *system state vector*, a column vector with n rows;
- $\mathbf{F}(t)$ is the *dynamic coefficient matrix*, an $n \times n$ matrix;
- $\mathbf{G}(t)$ is a *dynamic noise distribution matrix*, which can be an identity matrix; and
- $\mathbf{w}(t)$ is a zero-mean white-noise vector representing *dynamic disturbance noise*, also called *process noise*.

Example 10.1 Stochastic Differential Equation Model for Harmonic Resonator. Dynamical behavior of the one-dimensional damped mass–spring illustratedy in Fig. 10.6 is modeled by the equations

$$m\frac{d^2\xi}{dt^2} = ma = F = \underbrace{-C_{\text{damping}}\frac{d\xi}{dt}}_{\text{damping force}} - \underbrace{C_{\text{spring}}\xi}_{\text{spring force}} + \underbrace{w(t)}_{\text{disturbance}}$$

or

$$\frac{d^2\xi}{dt^2} + \frac{C_{\text{damping}}}{m}\frac{d\xi}{dt} + \frac{C_{\text{spring}}}{m}\xi = \frac{w(t)}{m}, \tag{10.46}$$

[7]Langevin was a prolific scientist with pioneering work in many areas, including para- and diamagnetism and sonar.

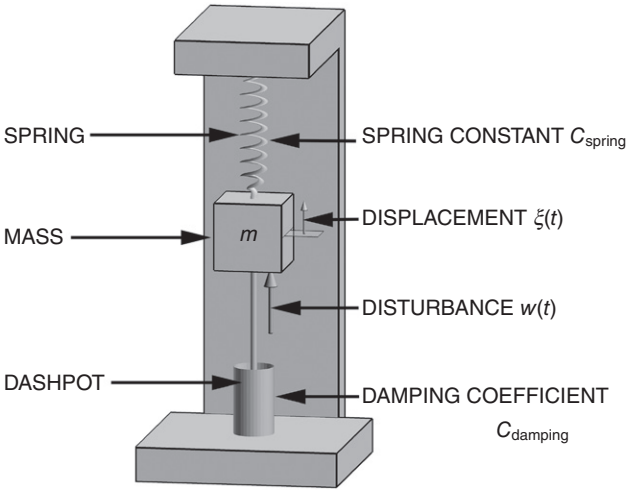


Fig. 10.6 Model for dynamic system of Example 10.1.

where

m is the mass attached to spring and damper,
ξ is the upward displacement of the mass from its rest position,
C_{spring} is the spring constant,
C_{damping} is the damping coefficient of the dashpot, and
w(t) is the random disturbing force acting on the mass.

10.3.1.3 Systems of First-Order Linear Differential Equations The so-called *state space models* for dynamic systems replace higher-order differential equations with systems of first-order differential equations. This can be done by defining the first $n - 1$ derivatives of an nth-order differential equation as state variables.

Example 10.2 State Space Model for Harmonic Resonator. Equation 10.46 is a linear second-order ($n = 2$) differential equation. It can be transformed into a system of two linear first-order differential equations with state variables

$$x_1 \overset{\text{def}}{=} \xi \,(\text{mass displacement}),$$

$$x_2 \overset{\text{def}}{=} \frac{d\xi}{dt} \,(\text{mass velocity}),$$

for which

$$\frac{dx_1}{dt} = x_2 \tag{10.47}$$

$$\frac{dx_2}{dt} = \frac{-C_{\text{spring}}}{m} x_1 + \frac{-C_{\text{damping}}}{m} x_2 + \frac{w(t)}{m}. \tag{10.48}$$

10.3.1.4 Representation in Terms of Vectors and Matrices State space models using systems of linear first-order differential equations can be represented more compactly in terms of a *state vector*, *dynamic coefficient matrix*, and *dynamic disturbance vector*.

Systems of linear first-order differential equations represented in "long-hand" form as

$$\begin{aligned}
\frac{dx_1}{dt} &= f_{11}x_1 + f_{12}x_2 + f_{13}x_3 + \cdots + f_{1n}x_n + w_1,\\
\frac{dx_2}{dt} &= f_{21}x_1 + f_{22}x_2 + f_{23}x_3 + \cdots + f_{2n}x_n + w_2,\\
\frac{dx_3}{dt} &= f_{31}x_1 + f_{32}x_2 + f_{33}x_3 + \cdots + f_{3n}x_n + w_3,\\
&\vdots \quad \vdots \quad \vdots \quad \vdots \quad \ddots \quad \vdots \quad \vdots\\
\frac{dx_n}{dt} &= f_{n1}x_1 + f_{n2}x_2 + f_{n3}x_3 + \cdots + f_{nn}x_n + w_n
\end{aligned}$$

can be represented more compactly in matrix form as

$$\frac{d}{dt}\mathbf{x} = \mathbf{F}\mathbf{x} + \mathbf{w}, \tag{10.49}$$

where the *state vector* $\mathbf{x}$, *dynamic coefficient matrix* $\mathbf{F}$, and *dynamic disturbance vector* $\mathbf{w}$ are given as

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}, \mathbf{F} = \begin{bmatrix} f_{11} & f_{12} & f_{13} & \cdots & f_{1n} \\ f_{21} & f_{22} & f_{23} & \cdots & f_{2n} \\ f_{13} & f_{23} & f_{33} & \cdots & f_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f_{n1} & f_{n2} & f_{n3} & \cdots & f_{nn} \end{bmatrix}, \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix},$$

respectively.

Example 10.3 Harmonic Resonator Model in Matrix Form. For the system of linear differential Eqs. 10.47 and 10.48,

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},$$

$$\mathbf{F} = \begin{bmatrix} 0 & 1 \\ -C_{\text{spring}} / m & -C_{\text{damping}} / m \end{bmatrix}$$

$$\mathbf{w}(t) = \begin{bmatrix} 0 \\ w(t) / m \end{bmatrix}.$$

10.3.1.5 Eigenvalues of Dynamic Coefficient Matrices The coefficient matrix $\mathbf{F}$ of a system of linear differential equations $\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{w}$ has effective units of reciprocal time or frequency (in units of radian per second). It is perhaps then not surprising that the characteristic values (eigenvalues) of $\mathbf{F}$ are the characteristic frequencies of the dynamic system represented by the differential equations.

The eigenvalues of an $n \times n$ matrix $\mathbf{F}$ are the roots $\{\lambda_i\}$ of its *characteristic polynomial*:

$$\det(\lambda \mathbf{I} - \mathbf{F}) = \sum_{k=0}^{n} a_n \lambda^n = 0. \tag{10.50}$$

The eigenvalues of $\mathbf{F}$ have the same interpretation as the poles of the related system transfer function, in that the dynamic system $\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{w}$ is stable if and only if the solutions of the characteristic equation $\det(\lambda \mathbf{I} - \mathbf{F}) = 0$ lie in the left half-plane.

Example 10.4 Damping and Resonant Frequency for Underdamped Harmonic Resonator. For the dynamic coefficient matrix

$$\mathbf{F} = \begin{bmatrix} 0 & 1 \\ -C_{\text{spring}} / m & -C_{\text{damping}} / m \end{bmatrix} \tag{10.51}$$

in Example 10.3, the eigenvalues of $\mathbf{F}$ are the roots of its characteristic polynomial

$$\begin{aligned} \det(\lambda \mathbf{I} - \mathbf{F}) &= \det \begin{bmatrix} \lambda & -1 \\ C_{\text{spring}} / m & \lambda + C_{\text{damping}} / m \end{bmatrix} \\ &= \lambda^2 + \frac{C_{\text{damping}}}{m} \lambda + \frac{C_{\text{spring}}}{m}, \end{aligned}$$

which are

$$\lambda = -\frac{C_{\text{damping}}}{2m} \pm \frac{1}{2m}\sqrt{C_{\text{damping}}^2 - 4mC_{\text{spring}}}.$$

If the discriminant

$$C_{\text{damping}}^2 - 4mC_{\text{spring}} < 0,$$

then the mass–spring system is called underdamped, and its eigenvalues are a complex conjugate pair:

$$\lambda = -\frac{1}{\tau_{\text{damping}}} \pm \omega_{\text{resonant}}\mathbf{i},$$

with real part

$$-\frac{1}{\tau_{\text{damping}}} = -\frac{C_{\text{damping}}}{2m}$$

and imaginary part

$$\omega_{\text{resonant}} = \frac{1}{2m}\sqrt{4mC_{\text{spring}} - C_{\text{damping}}^2}. \tag{10.52}$$

The alternative parameter

$$\tau_{\text{damping}} = \frac{2m}{C_{\text{damping}}}$$

is called the damping time constant of the system, and the other parameter, ω_{resonant}, is the resonant frequency in units of radian per second.

The dynamic coefficient matrix for the damped harmonic resonator model can also be expressed in terms of the resonant frequency and damping time constant as

$$\mathbf{F}_{\text{harmonic resonator}} = \begin{bmatrix} 0 & 1 \\ -\omega^2 - 1/\tau^2 & -2/\tau \end{bmatrix}. \tag{10.53}$$

So long as the damping coefficient $C_{\text{damping}} > 0$, the eigenvalues of this system will lie in the left half-plane. In that case, the damped mass–spring system is guaranteed to be stable.

10.3.1.6 Matrix Exponential Function The matrix exponential function is defined as

$$\exp(\mathbf{M}) \overset{\text{def}}{=} \sum_{k=0}^{\infty} \frac{1}{k!}\mathbf{M}^k \tag{10.54}$$

for square matrices **M**. The result is a square matrix of the same dimension as **M**.

This function has some useful properties:

1. The matrix $\mathbf{N} = \exp(\mathbf{M})$ is always invertible and $\mathbf{N}^{-1} = \exp(-\mathbf{M})$.
2. If **M** is antisymmetric (i.e., its matrix transpose $\mathbf{M}^{\mathrm{T}} = -\mathbf{M}$), then $\mathbf{N} = \exp(\mathbf{M})$ is an *orthogonal* matrix (i.e., its matrix transpose $\mathbf{N}^{\mathrm{T}} = \mathbf{N}^{-1}$).
3. The eigenvalues of $\mathbf{N} = \exp(\mathbf{M})$ are the (scalar) exponential functions of the eigenvalues of **M**.
4. If $\mathbf{M}(s)$ is an integrable function of a scalar s, then the derivative

$$\frac{d}{dt}\exp\left(\int^{t} \mathbf{M}(s)\,ds\right) = \mathbf{M}(t)\exp\left(\int^{t} \mathbf{M}(s)\,ds\right). \tag{10.55}$$

10.3.1.7 Forward Solution The forward solution of a differential equation is a solution in terms of initial conditions. The property of the matrix exponential function shown in Eq. 10.55 can be used to define the forward solution of Eq. 10.49 as

$$\mathbf{x}(t) = \exp\left(\int_{t_0}^{t} \mathbf{F}(s)\,ds\right)\left[\mathbf{x}(t_0) + \int_{t_0}^{t} \exp\left(-\int_{t_0}^{s} \mathbf{F}(r)\,dr\right)\mathbf{w}(s)\,ds\right], \tag{10.56}$$

where $\mathbf{x}(t_0)$ is the *initial value* of the state vector **x** for $t \geq t_0$.

10.3.1.8 Time-Invariant Systems If the dynamic coefficient matrix **F** of Eq. 10.49 does not depend on t (time), then the problem is called *time invariant*. In that case,

$$\int_{t_0}^{t} \mathbf{F}\,ds = (t - t_0)\mathbf{F} \tag{10.57}$$

and the forward solution

$$\mathbf{x}(t) = \exp[(t - t_0)\mathbf{F}]\left\{\mathbf{x}(t_0) + \int_{t_0}^{t} \exp[-(s - t_0)\mathbf{F}]\mathbf{w}(s)\,ds\right\}. \tag{10.58}$$

10.3.2 Stochastic Systems in Discrete Time

10.3.2.1 Zero-Mean White Gaussian Noise Sequences *A zero-mean white Gaussian* noise process in discrete time is a sequence of *independent* samples . . . , $\mathbf{w}_{k-1}$, $\mathbf{w}_k$, $\mathbf{w}_{k+1}$, . . . from a normal probability distribution $N(0, \mathbf{Q}_k)$ with zero mean and known finite covariances $\mathbf{Q}_k$. In Kalman filtering, it is not necessary (but not unusual) that the covariance of all samples be the same.

Sampling is called independent if the expected values of outer products

$$E\left\langle \mathbf{w}_i \mathbf{w}_j^{\mathrm{T}} \right\rangle = \begin{cases} 0, & i \neq j, \\ \mathbf{Q}_i, & i = j, \end{cases} \tag{10.59}$$

for all integer indices i and j of the random process.

Zero-mean white Gaussian noise sequences are the fundamental random processes used in Kalman filtering. However, it is *not* necessary that all noise sources in the modeled sensors and dynamic systems be zero-mean white Gaussian noise sequences. It is only necessary that they can be modeled in terms of such processes.

10.3.2.2 Gaussian Linear Stochastic Processes in Discrete Time A linear stochastic processes model in discrete time has the form

$$\mathbf{x}_k = \mathbf{\Phi}_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \tag{10.60}$$

where $\mathbf{w}_k$ is a zero-mean white Gaussian noise process with known covariances $\mathbf{Q}_k$ and the vector $\mathbf{x}$ represents the state of a dynamic system.

This model for "marginally random" dynamics is quite useful for representing physical systems (e.g., land vehicles, seacraft, aircraft) with zero-mean random disturbances (e.g., winds or currents). The state transition matrix (STM) $\mathbf{\Phi}_k$ represents the known dynamic behavior of the system, and the covariance matrices $\mathbf{Q}_k$ represent the unknown random disturbances. Together, they model the propagation of the necessary statistical properties of the state variable $\mathbf{x}$.

Example 10.5 Harmonic Resonator with White Acceleration Disturbance Noise. If the disturbance acting on the harmonic resonator of Examples 10.1–10.6 were zero-mean white acceleration noise with variance $\sigma^2_{\text{disturbance}}$ then its disturbance noise covariance matrix would have the form

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 \\ 0 & \sigma^2_{\text{disturbance}} \end{bmatrix}. \tag{10.61}$$

10.3.3 State Space Models for Discrete Time

Measurements are the outputs of sensors sampled at *discrete times* $\ldots < t_{k-1} < t_k < t_{k+1} < \ldots$. The Kalman filter uses these values to estimate the state of the associated dynamic systems at those discrete times.

If we let $\ldots, \mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{x}_{k+1}, \ldots$ be the corresponding state vector values of a linear dynamic system at those discrete times, then each of these values can be determined from the previous value by using Eq. 10.58 in the form

$$\mathbf{x}_k = \mathbf{\Phi}_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1}, \tag{10.62}$$

$$\mathbf{\Phi}_{k-1} \overset{\text{def}}{=} \exp\left(\int_{t_{k-1}}^{t_k} \mathbf{F}(s)\,ds\right), \tag{10.63}$$

$$\mathbf{w}_{k-1} \overset{\text{def}}{=} \mathbf{\Phi}_k \int_{t_{k-1}}^{t_k} \exp\left(-\int_{t_{k-1}}^{t_k} \mathbf{F}(s)\,ds\right)\mathbf{w}(t)\,dt. \tag{10.64}$$

Equation 10.62 is the discrete-time dynamic system model corresponding to the continuous-time dynamic system model of Eq. 10.49.

The matrix $\mathbf{\Phi}_{k-1}$ (defined in Eq. 10.63) in the discrete-time model (Eq. 10.62) is called a **state transition matrix** for the dynamic system defined by $\mathbf{F}$. Note that $\mathbf{\Phi}$ depends only on $\mathbf{F}$, and not on the dynamic disturbance function $\mathbf{w}(t)$.

The noise vectors $\mathbf{w}_k$ are the discrete-time analog of the dynamic disturbance function $\mathbf{w}(t)$. They depend upon their continuous-time counterparts $\mathbf{F}$ and $\mathbf{w}$.

Example 10.6 State Transition Matrix for the Harmonic Resonator Model. The underdamped harmonic resonator model of Example 10.4 has no time-dependent terms in its coefficient matrix (Eq. 10.51), making it a time-invariant model with state transition matrix

$$\mathbf{\Phi} = \exp(\Delta t\mathbf{F}) \tag{10.65}$$

$$= e^{-\Delta t/\tau}\begin{bmatrix} \cos(\omega\Delta t) + \sin(\omega\Delta t)/\omega\tau & \sin(\omega\Delta t)/\omega \\ -\left[\sin(\omega\Delta t)/\omega\tau^2\right]\left[1+\omega^2\tau^2\right] & \cos(\omega\Delta t) - \sin(\omega\Delta t)/\omega\tau \end{bmatrix}, \tag{10.66}$$

where

ω is the resonant frequency, ω_{resonant},
τ is the damping time constant, τ_{damping}, and
Δt is the discrete time step.

The eigenvalues of $\mathbf{F}$ were shown to be $-1/\tau_{\text{damping}} \pm \mathbf{i}\omega_{\text{resonant}}$, so the eigenvalues of $\mathbf{F}\Delta t$ will be $-\Delta t/\tau_{\text{damping}} \pm \mathbf{i}\Delta t\omega_{\text{resonant}}$ and the eigenvalues of $\mathbf{\Phi}$ will be

$$\exp\left(-\frac{\Delta t}{\tau_{\text{damping}}} \pm \mathbf{i}\omega_{\text{resonant}}\Delta t\right) = e^{-\Delta t/\tau}[\cos(\omega\Delta t) \pm \mathbf{i}\sin(\omega\Delta t)].$$

A discrete-time dynamic system will be stable only if the eigenvalues of $\mathbf{\Phi}$ lie inside the unit circle (i.e., $|\lambda\ell| < 1$).

10.3.4 Dynamic Disturbance Noise Distribution Matrices

A common noise source can disturb more than one independent component of the state vector representing a dynamic system. Forces applied to a rigid body, for example, can affect rotational dynamics as well as translational dynamics. This sort of coupling of common disturbance noise sources into different components of the state dynamics can be represented by using a *noise distribution matrix* $\mathbf{G}$ in the form

$$\frac{d}{dt}\mathbf{x} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{w}(t), \tag{10.67}$$

where the components of $\mathbf{w}(t)$ are the common disturbance noise sources and the matrix $\mathbf{G}$ represents how these disturbances are distributed among the state vector components.

The covariance of state vector disturbance noise will then have the form $\mathbf{G}\mathbf{Q}_w\mathbf{G}^{\mathrm{T}}$, where $\mathbf{Q}_w$ is the covariance matrix for the white-noise process $\mathbf{w}(t)$.

The analogous model in discrete time has the form

$$\mathbf{x}_k = \mathbf{\Phi}_{k-1}\mathbf{x}_{k-1} + \mathbf{G}_{k-1}\mathbf{w}_{k-1}, \tag{10.68}$$

where $\{\mathbf{w}_k\}$ is a zero-mean white-noise process in discrete time.

In either case (i.e., continuous or discrete time), it is possible to use the noise distribution matrix for noise scaling as well, so that the components of $\mathbf{w}_k$ can be independent, uncorrelated unit normal variates and the noise covariance matrix $\mathbf{Q}_w = \mathbf{I}$, the identity matrix.

10.3.5 Predictor Equations

The linear stochastic process model parameters $\mathbf{\Phi}$ and $\mathbf{Q}$ can be used to calculate how the discrete-time process variables $\boldsymbol{\mu}$ (mean) and $\mathbf{P}$ (covariance) evolve over time.

Using Eq. 10.60 and taking expected values,

$$\begin{aligned}\hat{\mathbf{x}}_k &= \boldsymbol{\mu}_k \\ &\overset{\text{def}}{=} E\langle \mathbf{x}_k \rangle \\ &= E\langle \boldsymbol{\Phi}_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1} \rangle \\ &= \boldsymbol{\Phi}_{k-1} \mathrm{E}\langle \mathbf{x}_{k-1} \rangle + E\langle \mathbf{w}_{k-1} \rangle \\ &= \boldsymbol{\Phi}_{k-1}\hat{\mathbf{x}}_{k-1} + 0 \\ &= \boldsymbol{\Phi}_{k-1}\hat{\mathbf{x}}_{k-1}, \end{aligned} \tag{10.69}$$

$$\begin{aligned}\mathbf{P}_k &\overset{\text{def}}{=} \mathrm{E}\left\langle (\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^{\mathrm{T}} \right\rangle \\ &= E\left\langle (\boldsymbol{\Phi}_{k-1}\hat{\mathbf{x}}_{k-1} - \boldsymbol{\Phi}_{k-1}\mathbf{x}_{k-1} - \mathbf{w}_{k-1})(\boldsymbol{\Phi}_{k-1}\hat{\mathbf{x}}_{k-1} - \boldsymbol{\Phi}_{k-1}\mathbf{x}_{k-1} - \mathbf{w}_{k-1})^{\mathrm{T}} \right\rangle \\ &= \boldsymbol{\Phi}_{k-1} \underbrace{\mathrm{E}\left\langle (\mathbf{x}_{k-1} - \mathbf{x}_{k-1})(\mathbf{x}_{k-1} - \mathbf{x}_{k-1})^{\mathrm{T}} \right\rangle}_{\mathbf{P}_{k-1}} \boldsymbol{\Phi}_{k-1}^{\mathrm{T}} + \underbrace{E\left\langle \mathbf{w}_{k-1}\mathbf{w}_{k-1}^{\mathrm{T}} \right\rangle}_{\mathbf{Q}_{k-1}} \\ &\quad + \text{terms with zero expected value} \\ &= \boldsymbol{\Phi}_{k-1}\mathbf{P}_{k-1}\boldsymbol{\Phi}_{k-1}^{\mathrm{T}} + \mathbf{Q}_{k-1}. \end{aligned} \tag{10.70}$$

Equations 10.69 and 10.70 are the essential predictor equations for Kalman filtering.

10.4 SUMMARY OF KALMAN FILTER EQUATIONS

10.4.1 Essential Equations

The complete equations for the Kalman filter are summarized in Table 10.1.

10.4.2 Common Terminology

The symbols used in Table 10.1 for the variables and parameters of the Kalman filter are essentially those used in the original paper by Kalman [16], and this notation is fairly common in the literature.

The following are some names commonly used for the symbols in Table 10.1:

- $\mathbf{H}$ is the *measurement sensitivity matrix* or *observation matrix*.
- $\mathbf{H}\hat{\mathbf{x}}_k(-)$ is the *predicted measurement*.
- $\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}_k(-)$, the difference between the measurement vector and the predicted measurement, is the *innovations vector*.

TABLE 10.1. Essential Kalman Filter Equations

Predictor (Time or Temporal Updates)			
Predicted state vector $\hat{\mathbf{x}}_k(-)$	=	$\boldsymbol{\Phi}_k\hat{\mathbf{x}}_{k-1}(+)$	Eq. 10.69
Predicted covariance matrix: $\mathbf{P}_k(-)$	=	$\boldsymbol{\Phi}_k\mathbf{P}_{k-1}(+)\boldsymbol{\Phi}_k^{\mathrm{T}} + \mathbf{Q}_{k-1}$	Eq. 10.70
Corrector (Measurement or Observational Updates)			
Kalman gain $\bar{\mathbf{K}}_k$	=	$\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}}\left(\mathbf{H}_k\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}} + \mathbf{R}_k\right)^{-1}$	Eq. 10.41
Corrected state estimate $\hat{\mathbf{x}}_k(+)$	=	$\hat{\mathbf{x}}_k(-) + \bar{\mathbf{K}}_k(\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_k(-))$	Eq. 10.42
Corrected covariance matrix $\mathbf{P}_k(+)$	=	$\mathbf{P}_k(-) - \bar{\mathbf{K}}_k\mathbf{H}_k\mathbf{P}_k(-)$	Eq. 10.43

$\bar{\mathbf{K}}$ is the *Kalman gain*.
$\mathbf{P}_k(-)$ is the *predicted* or *a priori* value of estimation covariance.
$\mathbf{P}_k(+)$ is the *corrected* or *a posteriori* value of estimation covariance.
$\mathbf{Q}_k$ is the *covariance of dynamic disturbance noise*.
$\mathbf{R}$ is the *covariance of sensor noise* or *measurement uncertainty*.
$\hat{\mathbf{x}}_k(-)$ is the *predicted* or *a priori* value of the estimated state vector.
$\hat{\mathbf{x}}_k(+)$ is the *corrected* or *a posteriori* value of the estimated state vector.
$\mathbf{z}$ is the *measurement vector* or *observation vector*.

10.4.3 Data Flow Diagrams

The matrix-level data flow of the Kalman filter implementation for a time-varying problem is diagrammed in Fig. 10.7, with the inputs shown on the left, the outputs (corrected estimates) on the right, and the symbol z^{-1} representing the unit delay operator.

The dashed lines in the figure enclose two computation loops. The top loop is the estimation loop, with the feedback gain (Kalman gain) coming from the bottom loop. The bottom loop implements the Riccati equation solution used to calculate the Kalman gain. This bottom loop runs "open loop," in that there is no feedback mechanism to stabilize it in the presence of roundoff errors. Numerical instability problems with the Riccati equation propagation loop were discovered soon after the introduction of the Kalman filter.

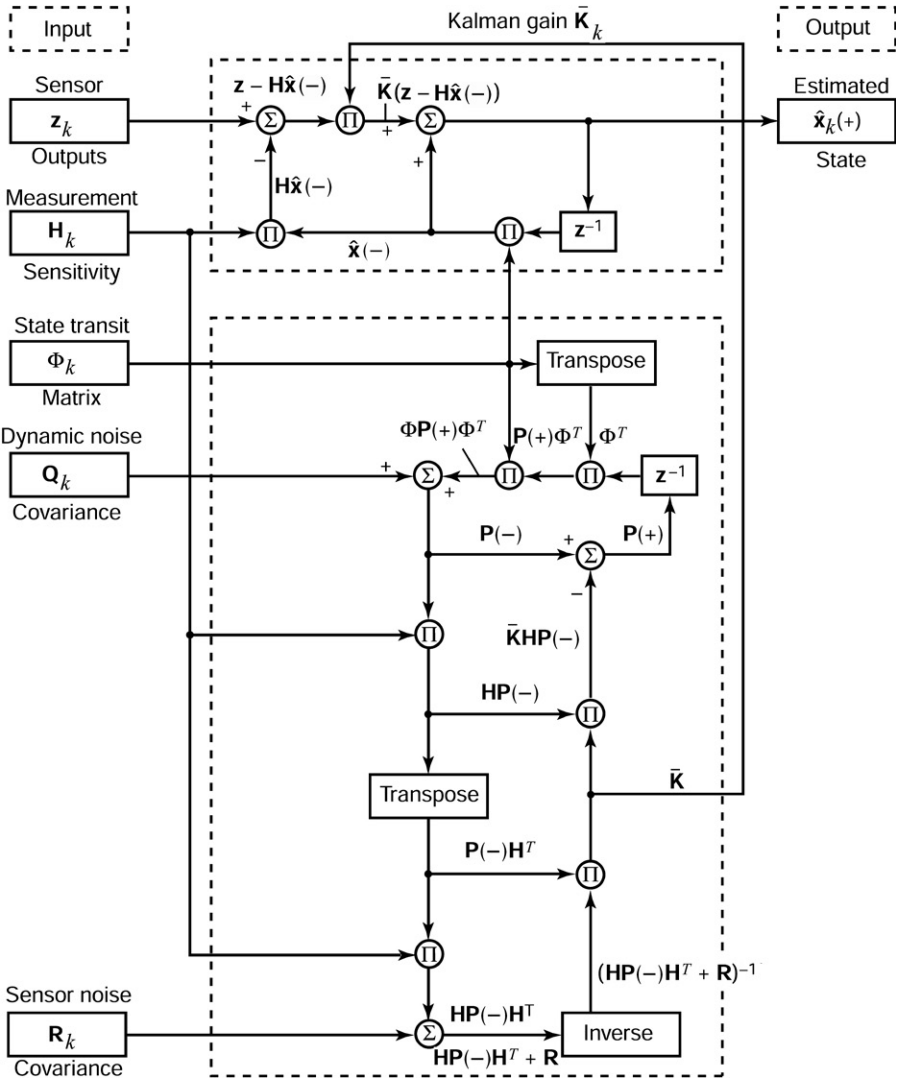


Fig. 10.7 Kalman filter data array flows for time-varying system.

10.5 ACCOMMODATING TIME-CORRELATED NOISE

The fundamental noise processes in the basic Kalman filter model are zero-mean white Gaussian noise processes $\{\mathbf{w}_k\}$, called *dynamic disturbance*, *plant noise*, or *process noise* and $\{\mathbf{v}_k\}$, called *sensor noise*, *measurement noise*, or *observation noise*.

GNSS signal propagation errors and INS position errors are not white-noise processes but are correlated over time. Fortunately, time-correlated noise processes can easily be accommodated in Kalman filtering by adding appropriate state variables to the Kalman filter model. A correlated noise process $\boldsymbol{\xi}_k$ can be modeled by a linear stochastic system model of the sort

$$\boldsymbol{\xi}_k = \boldsymbol{\Phi}_{k-1}\boldsymbol{\xi}_{k-1} + \mathbf{w}_{k-1}, \tag{10.71}$$

where $\{\mathbf{w}_k\}$ is a zero-mean white Gaussian noise process, then augment the state vector by appending the new variable $\boldsymbol{\xi}_k$,

$$\mathbf{x}_{\text{augmented}} = \begin{bmatrix} \mathbf{x}_{\text{original}} \\ \boldsymbol{\xi} \end{bmatrix}, \tag{10.72}$$

and modifying the parameter matrices $\boldsymbol{\Phi}$ and $\mathbf{Q}$, and modifying the parameter matrices $\boldsymbol{\Phi}$, $\mathbf{Q}$, and $\mathbf{H}$ accordingly.

10.5.1 Correlated Noise Models

10.5.1.1 Autocovariance Functions Correlation of a random sequence $\{\boldsymbol{\xi}\}$ is characterized by its discrete-time *autocovariance function* $\mathbf{P}_{\xi}[\Delta k]$, a function of the delay index Δk defined as

$$\mathbf{P}_{\xi}[\Delta k] \overset{\text{def}}{=} \underset{k}{\mathrm{E}}\left\langle (\boldsymbol{\xi}_k - \boldsymbol{\mu}_{\xi})(\boldsymbol{\xi}_{k+\Delta k} - \boldsymbol{\mu}_{\xi})^{\mathrm{T}} \right\rangle, \tag{10.73}$$

where $\boldsymbol{\mu}_{\xi}$ is the mean value of the random sequence $\{\boldsymbol{\xi}_k\}$.

For white-noise processes,

$$\mathbf{P}[\Delta k] = \begin{cases} 0, & \Delta k \neq 0, \\ \mathbf{Q}, & \Delta k = 0, \end{cases} \tag{10.74}$$

where $\mathbf{Q}$ is the covariance of the white-noise process.

10.5.1.2 Random Walks Random walks, also called *Wiener processes*, are cumulative sums of white-noise processes $\{\mathbf{w}_k\}$:

$$\boldsymbol{\xi}_k = \boldsymbol{\xi}_{k-1} + \mathbf{w}_{k-1}, \tag{10.75}$$

a stochastic process model with state transition matrix $\boldsymbol{\Phi} = \mathbf{I}$, an identity matrix.

Random walks are notoriously unstable, in the sense that the covariance of the variate $\boldsymbol{\xi}_k$ grows linearly with k and without bound as $k \to \infty$. In general, if any of the eigenvalues of a state transition matrix fall on or outside the unit

circle in the complex plane (as they all do for identity matrices), the variate of the stochastic process can fail to have a finite steady-state covariance matrix. However, as was demonstrated by R. E. Kalman in 1960, the covariance of *uncertainty* in the *estimated* system state vector can still converge to a finite steady-state value, even if the process itself is unstable.

10.5.1.3 Exponentially Correlated Noise Exponentially correlated random processes have finite, constant steady-state covariances. A scalar exponentially random process $\{\boldsymbol{\xi}_k\}$ has a model of the sort

$$\boldsymbol{\xi}_k = e^{-\Delta t/\tau}\boldsymbol{\xi}_{k-1} + \mathbf{w}_{k-1}, \tag{10.76}$$

where Δt is the time period between samples and τ is the exponential decay time constant of the process. The steady-state variance σ^2 of such a process is the solution to its steady-state variance equation,

$$\sigma_\infty^2 = e^{-2\Delta t/\tau}\sigma_\infty^2 + \mathbf{Q} \tag{10.77}$$

$$= \frac{\mathbf{Q}}{1 - e^{-2\Delta t/\tau}}, \tag{10.78}$$

where $\mathbf{Q}$ is the variance of the scalar zero-mean white-noise process $\{\mathbf{w}_k\}$.

The autocovariance sequence of an exponentially correlated random process in discrete time has the general form

$$P[\Delta k] = \sigma_\infty^2 \exp(-\Delta k / N_c), \tag{10.79}$$

which falls off exponentially on either side of its peak value σ_∞^2 (the process variance) at $\Delta k = 0$. The parameter N_c is called the *correlation number* of the process, where $N_c = \tau/\Delta t$ for correlation time τ and sample interval Δt.

10.5.1.4 Harmonic Noise Harmonic noise includes identifiable frequency components, such as those from AC power or from mechanical or electrical resonances. A stochastic process model for such sources has already been developed in the examples of this chapter.

10.5.1.5 Selective Availability (SA) At one time, the U.S. Department of Defense authorized deliberate pseudorandom dithering of GPS signal timing to derate navigation accuracies for nonmilitary users, when authorized military users had access to the correcting algorithm. The effect was largely negated by local broadcast of individual satellite timing errors determined by a receiver with known location. This "SA" dithering has ceased, but there is currently no guarantee that it will not be reinstated.

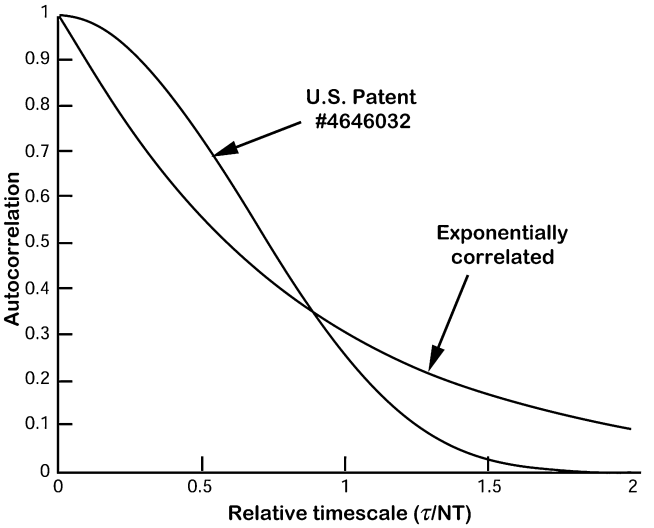


Fig. 10.8 Autocorrelation function.

The resulting SA timing errors have autocorrelation functions remarkably close to that shown in Fig. 10.8, which is from an algorithm patented by C. E. Wheatley III, C. G. Mosley, and E. V. Hunt, and issued as U. S. Patent No. 4,646,032, February 24, 1987, with the title *Controlled Oscillator Having Random Variable Frequency* [30]. The autocorrelation function of this pseudo-random dithering algorithm is published in the patent and is plotted in Fig. 10.8, along with an approximating exponential correlation function. Knowing the dithering algorithm does not give nonmilitary users any advantage other than being able to use its autocorrelation function in Kalman filtering of GPS signals. The correlation time of SA errors determined from GPS signal analysis is on the order of 10^2–10^3 s. It is also possible that the correlation times were being varied, which might explain the range of values reported in the literature.

10.5.1.6 Slow Variables Slowly varying error sources in GNSS/INS integration can include many of the calibration parameters of the inertial sensors, which can be responding to temperature variations or other unknown but slowly changing influences. The slow variations of these variables can often be tracked and compensated by combining the INS navigation estimates with the GNSS-derived estimates. What is different about the calibration parameters is that they are involved nonlinearly in the INS system model.

10.5.2 Empirical Modeling of Sensor Noise

Noise models used in Kalman filtering should be reasonably faithful representations of the true noise sources. Sensor noise can often be measured directly

and used in the design of an appropriate noise model. Dynamic process noise is not always so accessible, and its models must often be inferred from indirect measurements.

10.5.2.1 Spectral Characterization Spectrum analyzers and spectrum analysis software make it relatively easy to calculate the power spectral density of sampled noise data, and the results are useful for characterizing the type of noise and for identifying likely noise models.

The resulting noise models can then be simulated using pseudorandom sequences, and the power spectral densities of the simulated noise can be compared to that of the sampled noise to verify the model.

The power spectral density of white noise is constant across the spectrum, and each successive integral changes its slope by –20 dB/decade of frequency, as illustrated in Fig. 10.9.

10.5.2.2 Shaping Filters The spectrum of white noise is flat, and the amplitude spectrum of the output of a filter with white-noise input will have the shape of the amplitude transfer function of the filter, as illustrated in Fig. 10.10. Therefore, any noise spectrum can be approximated by white noise passed through a *shaping filter* to yield the desired shape. All correlated noise models for Kalman filters can be implemented by shaping filters.

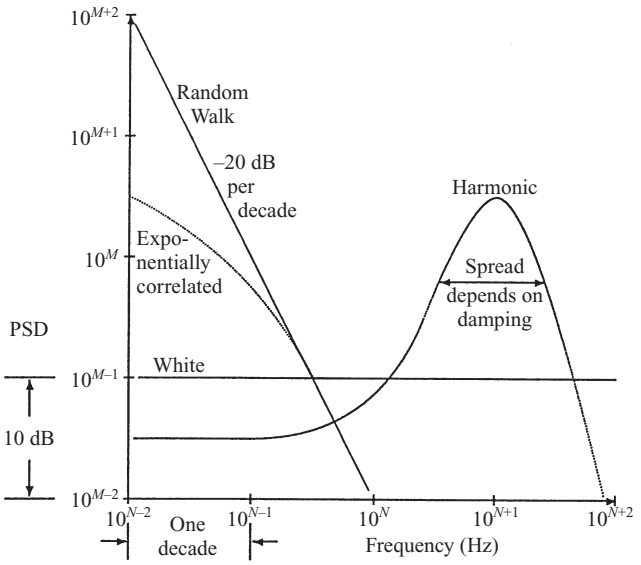


Fig. 10.9 Spectral properties of some common noise types. PSD, power spectral density.

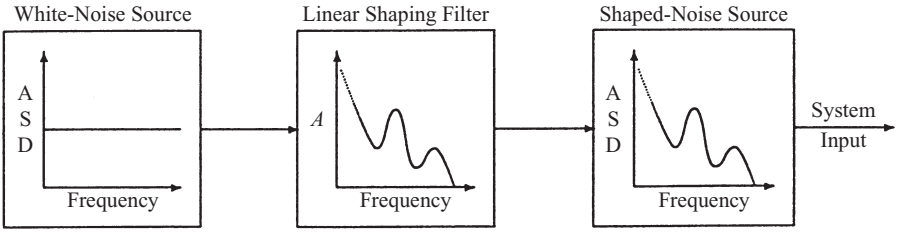


Fig. 10.10 Putting white noise through shaping filters. ASD, amplitude spectral density.

10.5.3 State Vector Augmentation

10.5.3.1 Correlated Dynamic Disturbance Noise A model for a linear stochastic process model in discrete time with uncorrelated and correlated disturbance noise has the form

$$\mathbf{x}_k = \mathbf{\Phi}_{x,k-1}\mathbf{x}_{k-1} + \mathbf{G}_{w_x,k-1}\mathbf{w}_{k-1} + \mathbf{D}_{\xi,k-1}\boldsymbol{\xi}_{k-1}, \tag{10.80}$$

where

$\mathbf{w}_{k-1}$ is the zero-mean white (i.e., uncorrelated) disturbance noise,
$\mathbf{G}_{w_x,k-1}$ is the white-noise distribution matrix,
$\boldsymbol{\xi}_{k-1}$ is zero-mean correlated disturbance noise, and
$\mathbf{D}_{\xi x,k-1}$ is the correlated noise distribution matrix.

If the correlated dynamic disturbance noise can be modeled as yet another linear stochastic process

$$\boldsymbol{\xi}_k = \mathbf{\Phi}_{\xi,k-1}\boldsymbol{\xi}_{k-1} + \mathbf{G}_{w_\xi,k-1}\mathbf{w}_{\xi,k-1} \tag{10.81}$$

with only zero-mean white-noise inputs $\{\mathbf{w}_{u,k}\}$, then the augmented state vector

$$\mathbf{x}_{\text{aug},k} \overset{\text{def}}{=} \begin{bmatrix} \mathbf{x}_k \\ \boldsymbol{\xi}_k \end{bmatrix} \tag{10.82}$$

has a stochastic process model

$$\mathbf{x}_{\text{aug},k} = \begin{bmatrix} \mathbf{\Phi}_{x,k-1} & \mathbf{D}_{\xi x,k-1} \\ 0 & \mathbf{\Phi}_{\xi,k-1} \end{bmatrix} x_{\text{aug},k-1} + \begin{bmatrix} \mathbf{G}_{w_x,k-1} & 0 \\ 0 & \mathbf{G}_{w_\xi,k-1} \end{bmatrix} \begin{bmatrix} \mathbf{w}_{x,k-1} \\ \mathbf{w}_{\xi,k-1} \end{bmatrix} \tag{10.83}$$

having only uncorrelated disturbance noise with covariance

$$\mathbf{Q}_{\text{aug},k-1} = \begin{bmatrix} \mathbf{Q}_{wx,k-1} & 0 \\ 0 & \mathbf{Q}_{w_\xi,k-1} \end{bmatrix}. \tag{10.84}$$

The new measurement sensitivity matrix for this augmented state vector will have the block form

$$\mathbf{H}_{\text{aug},k} = [\mathbf{H}_k \quad 0]. \tag{10.85}$$

The augmenting block is zero in this case because the uncorrelated noise source is dynamic disturbance noise, not sensor noise.

10.5.3.2 Correlated Sensor Noise The same sort of state augmentation can be done for correlated sensor noise $\{\boldsymbol{\xi}_k\}$,

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{A}_k\mathbf{v}_k + \mathbf{B}_k\boldsymbol{\xi}_k, \tag{10.86}$$

with the same type of model for the correlated noise (Eq. 10.81) and using the same augmented state vector (Eq. 10.82), but now with a different augmented state transition matrix,

$$\mathbf{F}_{\text{aug},k-1} = \begin{bmatrix} \mathbf{F}_{x,k-1} & 0 \\ 0 & \mathbf{F}_{\xi,k-1} \end{bmatrix}, \tag{10.87}$$

and augmented measurement sensitivity matrix,

$$\mathbf{H}_{\text{aug},k} = [\mathbf{H}_k \ \mathbf{B}_k]. \tag{10.88}$$

10.5.3.3 Correlated Noise in Continuous Time There is an analogous procedure for state augmentation using continuous-time models. If $\boldsymbol{\xi}(t)$ is a correlated noise source defined by a model of the sort

$$\frac{d}{dt}\boldsymbol{\xi} = \mathbf{F}_\xi\boldsymbol{\xi} + \mathbf{w}_\xi \tag{10.89}$$

for $\mathbf{w}_\xi(t)$, a white-noise source, then any stochastic process model of the sort

$$\frac{d}{dt}\mathbf{x} = \mathbf{F}_x\mathbf{x} + \mathbf{w}_x(t) + \boldsymbol{\xi}(t) \tag{10.90}$$

with this correlated noise source can also be modeled by the augmented state vector

$$\mathbf{x}_{\text{aug}} \overset{\text{def}}{=} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\xi} \end{bmatrix} \tag{10.91}$$

as

$$\frac{d}{dt}\mathbf{x}_{\text{aug}} = \begin{bmatrix} \mathbf{F}_x & \mathbf{I} \\ 0 & \mathbf{F}_\xi \end{bmatrix} \mathbf{x}_{\text{aug}} + \begin{bmatrix} \mathbf{w}_x \\ \mathbf{w}_\xi \end{bmatrix} \tag{10.92}$$

with only uncorrelated disturbance noise.

10.6 NONLINEAR AND ADAPTIVE IMPLEMENTATIONS

Although the Kalman filter is defined for linear dynamic systems with linear sensors, it has been applied more often than not to real-world applications without truly linear dynamics or sensors—and usually with remarkably great success.

The following subsections show how this has been done and how to assess the risk of serious linear approximation errors.

10.6.1 Assessing Linear Approximation Errors

Although the Kalman filter derivation assumes the dynamic and measurement models are linear, the resulting filter has been applied with impunity—and considerable success—to many nonlinear problems. The question is: How does one assess whether the linear approximation errors are acceptable?

An approach to implementing such a test is derived and demonstrated, showing that linearization of pseudorange measurement sensitivities by partial derivative approximation is statistically acceptable for Kalman filtering in GNSS receiver implementations.

10.6.1.1 Statistical Measures of Acceptability The essential idea is that, within reasonably expected variations of the state vector from its estimated value (as determined by the covariance of state estimation uncertainty), the mean-squared errors due to linearization should be dominated by the modeled uncertainties. For measurement nonlinearities, the modeled uncertainties are characterized by the measurement noise covariance $\mathbf{R}$. For dynamic nonlinearities, the model uncertainties are characterized by the dynamic disturbance noise covariance $\mathbf{Q}$.

The range of perturbations under which these conditions need to be met is generally determined by the magnitude of uncertainty in the estimate. The range can be specified in terms of the standard deviations of uncertainty.

The resulting statistical conditions for linearization can be stated in the following manner:

1. For the temporal state transition function $\boldsymbol{\phi}(\mathbf{x})$, the linear approximation error should be insignificant compared to $\mathbf{Q}$ when the state vector variations $\delta\mathbf{x}$ of $\hat{\mathbf{x}}$ are statistically significant. This condition can be met if the values of $\delta\mathbf{x}$ are smaller than the $N\sigma$-values of the estimated distribution

of uncertainty in the estimate $\hat{\mathbf{x}}$, which is characterized by the covariance matrix $\mathbf{P}$, for $N \geq 3$; that is, for

$$(\delta\mathbf{x})^T \mathbf{P}^{-1} (\delta\mathbf{x}) \leq N^2, \tag{10.93}$$

the linear approximation error

$$\boldsymbol{\varepsilon} \overset{\text{def}}{=} \boldsymbol{\phi}(\hat{\mathbf{x}} + \delta\mathbf{x}) - \left[\boldsymbol{\phi}(\hat{\mathbf{x}}) + \left. \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}} \delta\mathbf{x} \right] \tag{10.94}$$

should be bounded by

$$\boldsymbol{\varepsilon}^T \mathbf{Q}^{-1} \boldsymbol{\varepsilon} \ll 1; \tag{10.95}$$

that is, for the expected range of variation of estimation errors, the nonlinear approximation errors are dominated by modeled dynamic uncertainty.

2. For the measurement/sensor transformation $h(x)$: for $N\sigma \geq 3\sigma$ perturbations of $\hat{x}$, the linear approximation error should be insignificant compared to R, that is, for some $N \geq 3$, for all perturbations δx of $\hat{x}$ such that

$$(\delta\mathbf{x})^T \mathbf{P}^{-1} (\delta\mathbf{x}) \leq N^2, \tag{10.96}$$

$$\boldsymbol{\varepsilon}_h = \underbrace{\mathbf{h}(\hat{\mathbf{x}} + \delta\mathbf{x}) - \left[\mathbf{h}(\hat{\mathbf{x}}) + \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}} \delta\mathbf{x} \right]}_{\text{approximation error}}, \tag{10.97}$$

$$\boldsymbol{\varepsilon}_h^T \mathbf{R}^{-1} \boldsymbol{\varepsilon}_h \ll 1. \tag{10.98}$$

The value of estimation uncertainty covariance P used in Eq. 10.96 would ordinarily be the *a priori* value, calculated before the measurement is used. If the measurement update uses what is called the *iterated extended Kalman filter* (IEKF), however, the *a posteriori* value can be used.

Verifying these conditions requires simulating a nominal trajectory for $\mathbf{x}(t)$, implementing the Riccati equation to compute the covariance $\mathbf{P}$, sampling the estimate $\hat{\mathbf{x}} = \mathbf{x} + \delta\mathbf{x}$ to satisfy the test conditions, and evaluating the test conditions to verify that the problem is adequately quasilinear. The statistical parameter N is essentially a measure of confidence that the linear approximation will be insignificant in the actual application.

10.6.1.2 Sampling for Acceptability Testing As a minimum, the perturbations $\delta\mathbf{x}$ can be calculated along the principal axes on the $N\sigma$ equiprobability

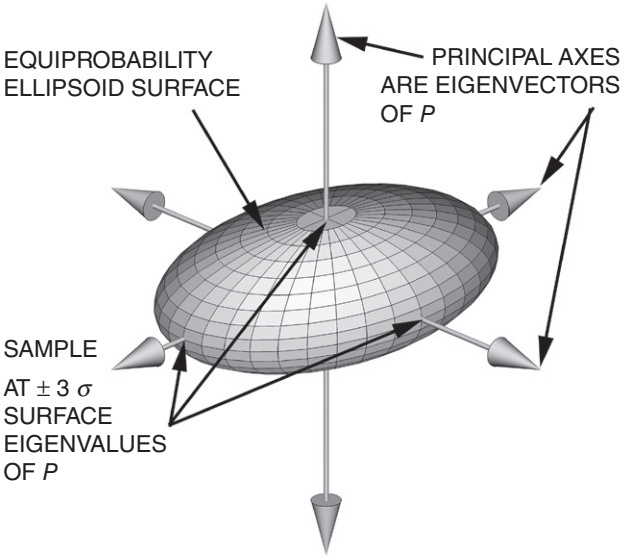


Fig. 10.11 Sampling along principal axes of 3σ equiprobability ellipsoid.

hyperellipse of the Gaussian distribution of estimation uncertainty. Figure 10.11 is an illustration of such an equiprobability ellipsoid in three dimensions, with arrows drawn along the principal axes of the ellipsoid. These axes and their associated 1σ values can be calculated in MATLAB® by using the singular value decomposition (SVD) of a covariance matrix $\mathbf{P}$:

$$\mathbf{P} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \tag{10.99}$$

$$= \sum_{i=1}^{n} \mathbf{u}_i \sigma_i^2 \mathbf{u}_i^T \tag{10.100}$$

$$\delta\mathbf{x}_i = N\sigma_i \mathbf{u}_i, 1 \le i \le n \tag{10.101}$$

$$\delta\mathbf{x}_{2n+i} = -N\sigma_i \mathbf{u}_i, 1 \le i \le n, \tag{10.102}$$

where the vectors $\mathbf{u}_i$ are the columns of the orthogonal matrix $\mathbf{U}$ in the SVD of $\mathbf{P}$. The principal standard deviations σ_i are the square roots of the diagonal elements of the matrix $\mathbf{\Lambda}$ in the SVD.

The procedure outlined by Eqs. 10.101 and 10.102 will yield $2n$ perturbation samples, where n is the dimension of x.

Conditions in Equations 10.95 and 10.98 depend on estimation uncertainty. The bigger the uncertainties in $\hat{\mathbf{x}}$ are, the larger the perturbations must be for satisfying the quasilinearity constraints.

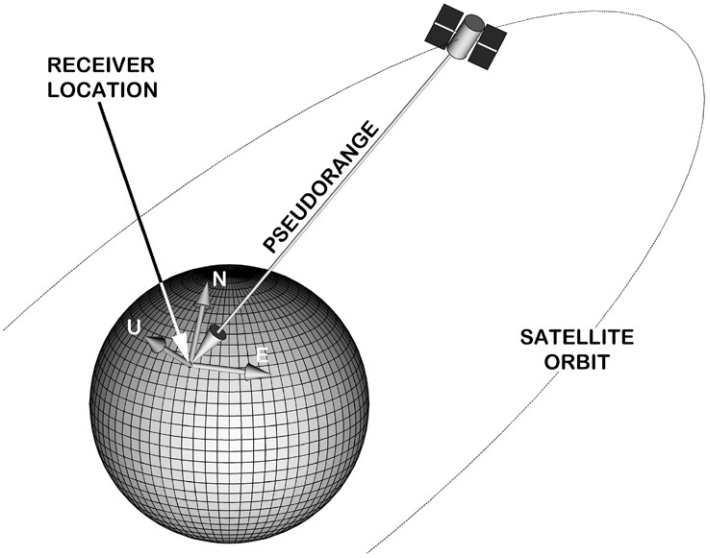


Fig. 10.12 GNSS satellite pseudorange ρ.

Example 10.7 Linearity of Satellite Pseudorange Measurements. GNSSs use satellites in orbit around the earth as radio navigation beacons. Receivers on or near the surface of the earth use their internal clocks to measure the time-delay Δt between when the signal was broadcast from each satellite in view, and when it was received. The product of c, the speed of propagation, times the time delay,

$$\rho = c\Delta t,$$

is called the pseudorange to the satellite from the receiver antenna. The essential geometric model for a single satellite is illustrated in Fig. 10.12.

The GNSS navigation solution for the location of the antenna requires several such pseudoranges to satellites in different directions. Each satellite broadcasts it precise location to the receiver, and the receiver processor is tasked to estimate the location of its antenna from these pseudoranges and satellite positions. This is usually done by extended Kalman filtering, using the derivatives of psuedoranges with respect to receiver location to approximate the measurement sensitivity matrix.

One can use Eq. 10.97 to determine whether the pseudorange measurement is sufficiently linear, given the uncertainty in the receiver antenna position, the nonlinear pseudorange measurement model, and the uncertainty in the pseudorange measurement.

For simplicity, we assume that the root mean square (RMS) position uncertainty is uniform in all directions, and

$$\mathbf{P} = \begin{bmatrix} \sigma_{\text{pos}}^2 & 0 & 0 \\ 0 & \sigma_{\text{pos}}^2 & 0 \\ 0 & 0 & \sigma_{\text{pos}}^2 \end{bmatrix} \text{(receiver position covariance)}, \tag{10.103}$$

$$R = \sigma_\rho^2 \text{ (pseudorange noise variance)}, \tag{10.104}$$

$$\sigma_\rho = 10 \text{ m (RMS pseudorange noise)}, \tag{10.105}$$

$$R_{\text{sat}} = 2.66 \times 10^7 \text{ m (satellite orbit radius)}, \tag{10.106}$$

$$R_{\text{rec}} = 6.378 \times 10^6 \text{ m (receiver radius from the earth center)}, \tag{10.107}$$

$$\alpha = 0, 30, 60, \text{and } 90^\circ \text{ elevation of satellite above the horizon.} \tag{10.108}$$

$$\rho_0 = \sqrt{R_{\text{sat}}^2 - R_{\text{rec}}^2 \cos(\alpha)^2} - R_{\text{rec}} \sin(\alpha) \text{ (nominal pseudorange)}, \tag{10.109}$$

$$\mathbf{X}_{\text{sat}} = \rho_0 \begin{bmatrix} -\cos(\alpha) \\ \sin(\alpha) \end{bmatrix} \text{(satellite position in east-north-up coordinates)} \tag{10.110}$$

$$\hat{\mathbf{x}} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \text{(estimated receiver position in east-north-up coordinates)}, \tag{10.111}$$

$$\mathbf{X}_{\text{rec}} = \delta\mathbf{x}, \tag{10.112}$$

$$= \begin{bmatrix} \delta x_{\text{E}} \\ \delta x_{\text{N}} \\ \delta x_{\text{U}} \end{bmatrix} \text{(true receiver position in east-north-up coordinates)}, \tag{10.113}$$

$$\rho = |\mathbf{X}_{\text{rec}} - \mathbf{X}_{\text{sat}}| \text{ (true pseudorange)}, \tag{10.114}$$

$$= \mathbf{h}(\hat{\mathbf{x}} + \delta\mathbf{x}), \tag{10.115}$$

where Eq. 10.114 is the nonlinear formula for pseudorange as a function of perturbations $\delta\mathbf{x}$ of the receiver antenna position in the east–north–up coordinates.

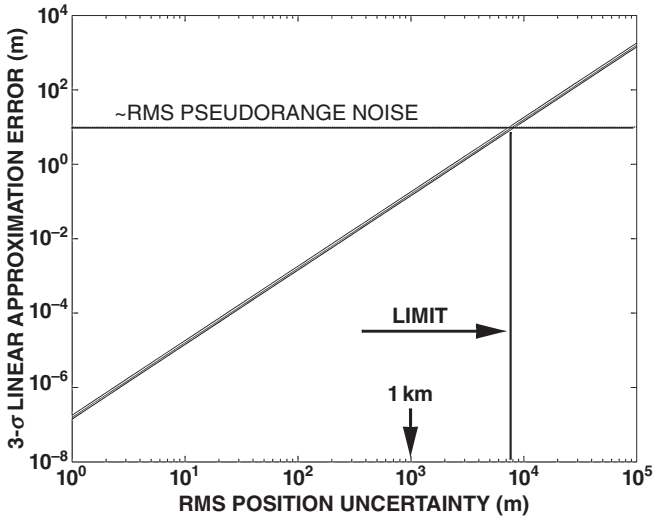


Fig. 10.13 Linearization error analysis of pseudorange measurements.

The linear approximation for the senstivity of pseudorange to antenna position is

$$\mathbf{H} \approx \left.\frac{\partial h}{\partial \delta \mathbf{x}}\right|_{\delta \mathbf{x}=0} \tag{10.116}$$

$$= [\cos(\alpha) \quad 0 \quad -\sin(\alpha)]. \tag{10.117}$$

In this case, the nonlinear approximation error defined by Eq. 10.97 will be a function of the satellite elevation angle α and the perturbation δx.

The RMS nonlinearity error for the six perturbations

$$\delta \mathbf{x} = \begin{bmatrix} \pm 3\sigma_{\text{pos}} \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \pm 3\sigma_{\text{pos}} \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ \pm 3\sigma_{\text{pos}} \end{bmatrix} \tag{10.118}$$

is plotted in Fig. 10.13 as a function of σ_{pos} for elevation angles $\alpha = 0°, 30°, 60°$, and 90° above the horizon. The four barely distinguishable solid diagonal lines in the plot are for these four different satellite elevation angles, which have little influence on nonlinearity errors. The dashed horizontal line represents the RMS pseudorange noise, indicating that nonlinear approximation errors are dominated by pseudorange noise for RMS position uncertainties less than ≈7 km.

Typical RMS position errors in GNSS navigation are in the order of 1–100 m, which is well within the quasilinear range. This indicates that extended Kalman filtering is certainly justified for GNSS navigation applications.

10.6.2 Nonlinear Dynamics

State dynamics for nonlinear systems can be expressed in the functional form

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}, t) + \mathbf{w}(t). \tag{10.119}$$

For this to be linearized, the function $\mathbf{f}$ must be differentiable, with Jacobian matrix

$$\mathbf{F}(\mathbf{x}, t) \overset{\text{def}}{=} \underbrace{\left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}(t)}}_{\text{EXTENDED}} \quad \text{or} \quad \underbrace{\left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}_{\text{nominal}}(t)}}_{\text{LINEARIZED}}, \tag{10.120}$$

where the extended Kalman filter uses the estimated trajectory for evaluating the Jacobian, and linearized Kalman filtering uses a nominal trajectory $\mathbf{x}_{\text{nominal}}(t)$, which may come from a simulation.

10.6.2.1 Nonlinear Dynamics with Control In applications with control variables $\mathbf{u}(t)$, Eq. 10.119 can also be expressed in the form

$$\frac{d}{dt}\mathbf{x} = \mathbf{f}[\mathbf{x}, \mathbf{u}(t), t] + \mathbf{w}(t), \tag{10.121}$$

in which case the control vector $\mathbf{u}$ may also appear in the Jacobian matrix $\mathbf{F}$.

10.6.2.2 Propagating Estimates The estimate $\hat{\mathbf{x}}$ is propagated by solving the differential equation

$$\frac{d}{dt}\hat{\mathbf{x}} = \mathbf{f}(\hat{\mathbf{x}}, t), \tag{10.122}$$

using whatever means necessary (e.g., Runge–Kutta integration). The solution is called the *trajectory* of the estimate.

10.6.2.3 Propagating Covariances The covariance matrix for nonlinear systems is also propagated over time as the solution to the matrix differential equation

$$\frac{d}{dt}\mathbf{P}(t) = \mathbf{F}(\mathbf{x}(t), t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}^{\mathrm{T}}(\mathbf{x}(t), t) + \mathbf{Q}(t), \tag{10.123}$$

where the values of $\mathbf{F}(x, t)$ from Eq. 10.123 must be calculated along a trajectory $\mathbf{x}(t)$. This trajectory can be the solution for the estimated value $\hat{\mathbf{x}}$ calculated

using the Kalman filter and Eq. 10.120 (for the extended Kalman filter) or along any "nominal" trajectory (for the "linearized" Kalman filter).

10.6.3 Nonlinear Sensors

Nonlinear Kalman filtering can accommodate sensors that are not truly linear but can at least be represented in the functional form

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k, \tag{10.124}$$

where $\mathbf{h}$ is a smoothly differentiable function of $\mathbf{x}$. For example, even linear sensors with nonzero biases (output offsets) $\mathbf{b}_{\text{sensor}}$ will have sensor models of the sort

$$\mathbf{h}(\mathbf{x}) = \mathbf{H}\mathbf{x} + \mathbf{b}_{\text{sensor}}, \tag{10.125}$$

in which case the Jacobian matrix

$$\frac{\partial \mathbf{h}}{\partial \mathbf{x}} = \mathbf{H}. \tag{10.126}$$

10.6.3.1 Predicted Sensor Outputs The predicted value of nonlinear sensor outputs uses the full nonlinear function applied to the estimated state vector:

$$\hat{\mathbf{z}}_k = \mathbf{h}_k(\hat{\mathbf{x}}_k). \tag{10.127}$$

10.6.3.2 Calculating Kalman Gains The value of the measurement sensitivity matrix $\mathbf{H}$ used in calculating the Kalman gain is evaluated as a Jacobian matrix,

$$\mathbf{H}_k = \underbrace{\left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}}}_{\text{EXTENDED}} \quad \text{or} \quad \underbrace{\left.\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{\text{nominal}}}}_{\text{LINEARIZED}}, \tag{10.128}$$

where the first value (used for extended Kalman filtering) uses the estimated trajectory for evaluation of partial derivatives, and the second value uses a nominal trajectory (used for the linearized Kalman filtering).

10.6.4 Linearized Kalman Filter

Perhaps the simplest approach to Kalman filtering for nonlinear systems uses linearization of the system model about a nominal trajectory. This approach is necessary for preliminary analysis of systems during the system design phase, when there may be several potential trajectories defined by different mission

TABLE 10.2. Linearized Kalman Filter Equations

Predictor (Time Updates)			
Predicted state vector:			
$\hat{\mathbf{x}}_k(-)$	=	$\hat{\mathbf{x}}_{k-1}(+)+\int_{t_{k-1}}^{t_k} f(\hat{\mathbf{x}},t)dt$	Eq. 10.119
Predicted covariance matrix:			
$\dot{\mathbf{P}}$	=	$\mathbf{FP}+\mathbf{PF}^{\mathrm{T}}+\mathbf{Q}(t)$	Eq. 10.123
$\mathbf{F}$	=	$\left.\dfrac{\partial \mathbf{f}}{\partial \mathbf{x}}\right\|_{\mathbf{x}=\mathbf{x}_{\mathrm{nom}}(t)}$	Eq. 10.120
or			
$\mathbf{P}_k(-)$	=	$\boldsymbol{\Phi}_k\mathbf{P}_{k-1}(+)\boldsymbol{\Phi}_k^{\mathrm{T}}+\mathbf{Q}_{k-1}$	Eq. 10.70
Corrector (Measurement Updates)			
Kalman gain			
$\bar{\mathbf{K}}_k$	=	$\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}}\left[\mathbf{H}_k\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}}+\mathbf{R}_k\right]^{-1}$	Eq. 10.41
$\mathbf{H}_k$	=	$\left.\dfrac{\partial \mathbf{h}}{\partial \mathbf{x}}\right\|_{\mathbf{x}=\mathbf{x}_{\mathrm{nom}}}$	Eq. 10.128
Corrected state estimate			
$\hat{\mathbf{x}}_k(+)$	=	$\hat{\mathbf{x}}_k(-)+\bar{\mathbf{K}}_k[\mathbf{z}_k-\mathbf{h}_k(\hat{\mathbf{x}}_k(-))]$	Eq. 10.42 and Eq. 10.127
Corrected covariance matrix			
$\mathbf{P}_k(+)$	=	$\mathbf{P}_k(-)-\bar{\mathbf{K}}_k\mathbf{H}_k\mathbf{P}_k(-)$	Eq. 10.43

scenarios. The essential implementation equations for this case are summarized in Table 10.2.

10.6.5 Extended Kalman Filtering (EKF)

This approach is due to Stanley F. Schmidt, and it has been used successfully in an enormous number of nonlinear applications. It is a form of nonlinear Kalman filtering with all Jacobian matrices (i.e., $\mathbf{H}$ and/or $\mathbf{F}$) evaluated at $\hat{\mathbf{x}}$, the estimated state. The essential extended Kalman filter equations are summarized in Table 10.3, the major differences from the conventional Kalman filter equations of Table 10.1 being

1. integration of the nonlinear integrand $\dot{\mathbf{x}}=\mathbf{f}(\mathbf{x})$ to predict $\hat{\mathbf{x}}_k(-)$,
2. use of the nonlinear function $\mathbf{h}_k(\hat{\mathbf{x}}_k(-))$ in measurement prediction,
3. use of the Jacobian matrix of the dynamic model function $\mathbf{f}$ as the dynamic coefficient matrix $\mathbf{F}$ in the propagation of the covariance matrix, and
4. use of the Jacobian matrix of the measurement function $\mathbf{h}$ as the measurement sensitivity matrix $\mathbf{H}$ in the covariance correction and Kalman gain equations.

TABLE 10.3. Extended Kalman Filter Equations

Predictor (Time Updates)			
Predicted state vector			
$\hat{\mathbf{x}}_k(-)$	=	$\hat{\mathbf{x}}_{k-1}(+)+\int_{t_{k-1}}^{t_k}\mathbf{f}(\hat{\mathbf{x}},t)dt$	Eq. 10.119
Predicted covariance matrix			
$\dot{\mathbf{P}}$	=	$\mathbf{FP}+\mathbf{PF}^{\mathrm{T}}+\mathbf{Q}(t)$	Eq. 10.123
$\mathbf{F}$	=	$\left.\dfrac{\partial \mathbf{f}}{\partial \mathbf{x}}\right\|_{\mathbf{x}=\hat{\mathbf{x}}(t)}$	Eq. 10.120
or			
$\mathbf{P}_k(-)$	=	$\mathbf{\Phi}_k\mathbf{P}_{k-1}(+)\mathbf{\Phi}_k^{\mathrm{T}}+\mathbf{Q}_{k-1}$	Eq. 10.70
Corrector (Measurement Updates)			
Kalman gain			
$\bar{\mathbf{K}}_k$	=	$\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}}\left[\mathbf{H}_k\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}}+\mathbf{R}_k\right]^{-1}$	Eq. 10.41
$\mathbf{H}_k$	=	$\left.\dfrac{\partial \mathbf{h}}{\partial \mathbf{x}}\right\|_{\mathbf{x}=\hat{\mathbf{x}}}$	Eq. 10.128
Corrected state estimate			
$\hat{\mathbf{x}}_k(+)$	=	$\hat{\mathbf{x}}_k(-)+\bar{\mathbf{K}}_k[\mathbf{z}_k-\mathbf{h}_k(\hat{\mathbf{x}}_k(-))]$	Eq. 10.42 and Eq. 10.127
Corrected covariance matrix			
$\mathbf{P}_k(+)$	=	$\mathbf{P}_k(-)-\bar{\mathbf{K}}_k\mathbf{H}_k\mathbf{P}_k(-)$	Eq. 10.43

10.6.6 Adaptive Kalman Filtering

In adaptive Kalman filtering, nonlinearities in the model arise from making parameters of the model into functions of state variables. For example, the time constant τ of a scalar exponentially correlated process

$$x_k = \exp(-\Delta t/\tau)\,x_{k-1}+w_k$$

may be unknown or slowly time-varying, in which case it can be made part of the augmented state vector

$$\hat{\mathbf{x}}_{\text{aug}}=\begin{bmatrix}\hat{x}\\ \hat{\tau}\end{bmatrix}$$

with state transition matrix

$$\mathbf{\Phi}=\begin{bmatrix}\exp(-\Delta t/\hat{\tau}) & \Delta t\exp(-\Delta t/\hat{\tau})\,\hat{x}/\hat{\tau}^2\\ 0 & \exp(-\Delta t/\tau^*)\end{bmatrix},$$

where $\tau^* \gg \tau$ is the correlation time constant of the variations in $\hat{\tau}$.

Example 10.8 Tracking Time-Varying Frequency and Damping. Consider the problem of tracking the phase components of a damped harmonic oscillator with slowly time-varying resonant frequency and damping time constant. The state variables for this nonlinear dynamic system are

x_1, the in-phase component of the oscillator output signal (i.e., the only observable component);
x_2, the quadrature-phase component of the signal;
x_3, the damping time constant of the oscillator (nominally 5 s); and
x_4, the frequency of oscillator (nominally 2 πrad/s, or 1 Hz).

The dynamic coefficient matrix will be

$$\mathbf{F} = \begin{bmatrix} -1/x_3 & x_4 & x_1/x_3^2 & x_2 \\ -x_4 & -1/x_3 & x_2/x_3^2 & -x_1 \\ 0 & 0 & -1/\tau_\tau & 0 \\ 0 & 0 & 0 & -1/\tau_\omega \end{bmatrix},$$

where τ_τ is the correlation time for the time-varying oscillator damping time constant, and τ_ω is the correlation time for the time-varying resonant frequency of the oscillator.

If only the in-phase component or the oscillator output can be sensed, then the measurement sensitivity matrix will have the form

$$\mathbf{H} = [1 \quad 0 \quad 0 \quad 0].$$

Figure 10.14 is a sample output of the MATLAB® m-file `osc_ekf.m` on the accompanying website, which implements this extended Kalman filter. Note that it tracks the phase, amplitude, frequency, and damping of the oscillator.

The unknown or time-varying parameters can also be in the measurement model. For example, a sensor output with time-varying scale factor S and bias b can be modeled by the nonlinear equation $z = Sx + b$ and linearized using augmented state vector

$$\mathbf{x}_{\text{aug}} = \begin{bmatrix} x \\ S \\ b \end{bmatrix}$$

and measurement sensitivity matrix

$$\mathbf{H} = \begin{bmatrix} \hat{S} & \hat{x} & 1 \end{bmatrix}.$$

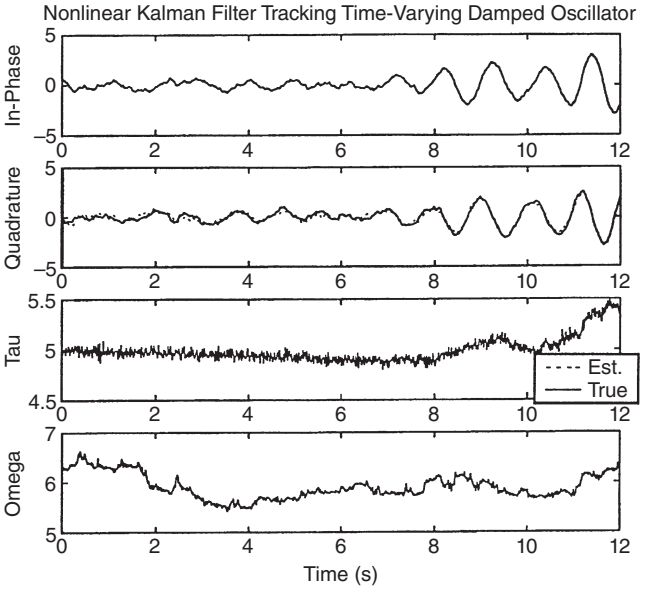


Fig. 10.14 Extended Kalman filter tracking simulated time-varying oscillator.

10.7 KALMAN–BUCY FILTER

The discrete-time form of the Kalman filter is well suited for computer implementation but is not particularly natural for engineers who find it more natural to think about dynamic systems in terms of differential equations.

The analog of the Kalman filter in continuous time is the Kalman–Bucy filter, developed jointly by Richard Bucy[8] and Rudolf Kalman [17].

10.7.1 Implementation Equations

The fundamental equations of the Kalman–Bucy filter are shown in Table 10.4.

People already familiar with differential equations may find the Kalman–Bucy filter more intuitive and easier to work with than the Kalman filter—despite complications of the stochastic calculus. To its credit, the Kalman–Bucy filter requires only one equation each for propagation of the estimate and its covariance, whereas the Kalman filter requires two (for prediction and correction).

However, if the result must eventually be implemented in a digital processor, then it will have to be put into discrete-time form. Formulas for

[8]Bucy recognized the covariance equation as a form of the nonlinear differential equation studied by Jacopo Francesco Riccati [25] (1676–1754), and that the equation was equivalent to spectral factorization in the Wiener filter.

TABLE 10.4. Kalman–Bucy Filter Equations

State Equation (Unified Predictor/Corrector)		
$\frac{d}{dt}\hat{\mathbf{x}}(t)$	=	$\mathbf{F}(t)\hat{\mathbf{x}}(t) + \underbrace{\mathbf{P}(t)\mathbf{H}^{\mathrm{T}}(t)\mathbf{R}^{-1}(t)}_{\bar{\mathbf{K}}_{\mathrm{KB}}(t)}[\mathbf{z}(t) - \mathbf{H}(t)\mathbf{x}(t)]$
Covariance Equation (Unified Predictor/Corrector)		
$\dot{\mathbf{P}}(t)$	=	$\mathbf{F}(t)\mathbf{P}(t) + \mathbf{P}(t)\mathbf{F}^{\mathrm{T}}(t) + \mathbf{Q}(t) - \bar{\mathbf{K}}_{\mathrm{KB}}(t)\mathbf{R}(t)\bar{\mathbf{K}}_{\mathrm{KB}(t)}^{\mathrm{T}}$

this transformation are given below. Those who prefer to "think in continuous time" can develop the problem solution first in continuous time as a Kalman–Bucy filter, then transform the result to Kalman filter form for implementation.

10.7.2 Kalman–Bucy Filter Parameters

Formulas for the Kalman filter parameters $\mathbf{Q}_k$ and $\mathbf{R}_k$ as functions of the Kalman–Bucy filter parameters $\mathbf{Q}(t)$ and $\mathbf{R}(t)$ can be derived from the process models.

$\mathbf{Q}(t)$ and $\mathbf{Q}_k$

The relationship between these two distinct matrix parameters depends on the coefficient matrix $\mathbf{F}(t)$ in the stochastic system model:

$$\mathbf{Q}_k = \int_{t_{k-1}}^{t_k} \exp\left(\int_t^{t_k} \mathbf{F}(s)\,ds\right)\mathbf{Q}(t)\exp\left(\int_t^{t_k} \mathbf{F}(s)\,ds\right)^{\mathrm{T}} dt. \tag{10.129}$$

$\mathbf{R}(t)$ and $\mathbf{R}_k$

This relationship will depend on how the sensor outputs in continuous time are filtered before sampling for the Kalman filter. If the sensor outputs were simply sampled without filtering, then

$$\mathbf{R}_k = \mathbf{R}(t_k). \tag{10.130}$$

However, it is common practice to use antialias filtering of the sensor outputs before sampling for Kalman filtering. Filtering of this sort can also alter the parameter $\mathbf{H}$ between the two implementations. For an integrate-and-hold filter (an effective antialiasing filter), this relationship has the form

$$\mathbf{R}_k = \int_{t_{k-1}}^{t_k} \mathbf{R}(t)\,dt, \tag{10.131}$$

in which case the measurement sensitivity matrix for the Kalman filter will be $\mathbf{H}_{\mathrm{K}} = \Delta t\mathbf{H}_{\mathrm{KB}}$, where $\mathbf{H}_{\mathrm{KB}}$ is the measurement sensitivity matrix for the Kalman–Bucy filter.

10.8 HOST VEHICLE TRACKING FILTERS FOR GNSS

Kalman filter models for estimating GNSS receiver antenna position and clock bias may include higher derivatives of position as additional state variables. One effect of adding state variables is the dilution of available pseudorange information, a result of which is increased position uncertainty with increased host vehicle dynamic activity. In Section 10.8.2, we quantify the relationship between dynamic uncertainty and position estimation uncertainty. In Section 10.8.3, we expand the modeling to include filters optimized for specific classes of host vehicle trajectories. In Section 10.8.4, we compare the relative efficacy of various filters on a specific application.

10.8.1 Vehicle Tracking Filters

Starting around 1950 in the United States, radar systems were integrated with computers to detect and track Soviet aircraft that might invade the continental United States [24]. The computer software included filters to identify and track individual aircraft within a formation. These "tracking filters" generated estimates of position and velocity for each aircraft, and they could be tuned to follow the unpredictable maneuvering capabilities of Soviet bombers of that era.

The same sorts of tracking filters are used in GNSS receivers to estimate the position and velocity of GNSS antennas on host vehicles with unpredictable dynamics. Important issues in the design and implementation of these filters include the following:

1. In what ways does vehicle motion affect GNSS navigation performance?
2. Which characteristics of vehicle motions influence the choice of tracking filter models?
3. How do we determine these characteristics for a specified vehicle type?

These issues are addressed in the following subsections.

10.8.2 Dynamic Dilution of Information

In addition to the "dilution of precision" (DOP) related to satellite geometry, there is a GNSS receiver "dilution of information" problem related to vehicle dynamics. In essence, if more information (in the measurements) is required to make up for the uncertainty of vehicle movement, then less information is left over for determining the instantaneous antenna position and clock bias.

For example, the location of a receiver antenna at a fixed position on the earth can be specified by three unknown constants (i.e., position coordinates

in three dimensions). Over time, as more and more measurements are used, the accuracy of the estimated position should improve. If the vehicle is moving, however, only the more recent measurements relate to the current antenna position.

10.8.2.1 Effect on Position Uncertainty Figure 10.15 is a plot of the contribution vehicle dynamic characteristics make to GNSS position estimation uncertainty for a range of host vehicle dynamic capabilities. In order to indicate the contributions that vehicle dynamics make to position uncertainty, this demonstration assumes that other contributory error sources are either negligible or nominal, for example,

- no receiver clock bias (that will come later);
- 10-m RMS time-correlated pseudorange error due to iono delay, receiver bias, interfrequency biases, and so on;
- 60-s pseudorange error correlation time;
- pseudoranges of each available satellite sampled every second;
- 10-RMS pseudorange uncorrelated measurement noise;
- 29-satellite GNSS configuration of March 8, 2006;
- only those satellites more than 15° above the horizon were used;
- 200-m/s RMS host vehicle velocity, representing a high-performance aircraft or missile;
- host vehicle at 40° north latitude; and
- results averaged over 1 h of simulated operation.

Figure 10.15 is output from the MATLAB® m-file `Damp2eval.m` on the accompanying website. It performs a set of GNSS tracking simulations using the "DAMP2" tracking filter described in Table 10.5 and in Section 10.8.3. This filter allows the designer to specify the RMS velocity, RMS acceleration, and acceleration correlation time of the host vehicle, and the plot shows how these two dynamic characteristics influence position estimation accuracy.

These results would indicate that navigation performance is more sensitive to vehicle acceleration magnitude than to its correlation time. Five orders-of-magnitude variation in correlation time do not cause one order-of-magnitude variation in RMS position estimation accuracy. At short correlation times, five orders-of-magnitude variation in RMS acceleration[9] cause around three orders-of-magnitude variation in RMS position estimation accuracy. These simulations were run at a 40° latitude. Changing simulation conditions may change the results somewhat.

The main conclusion is that unpredictable vehicle motion does, indeed, compromise navigation accuracy.

[9]The RMS acceleration used here does not include the acceleration required to counter gravity.

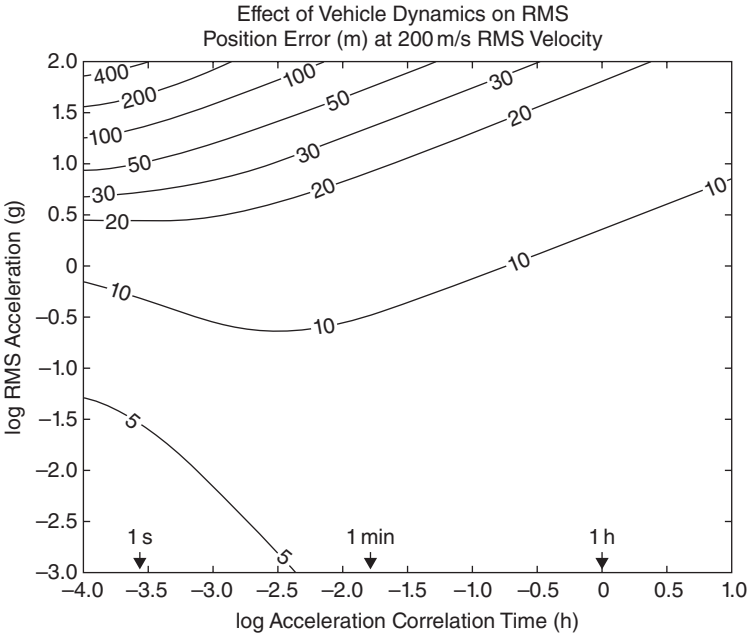


Fig. 10.15 DAMP2 tracker performance versus σ_{acc} and τ_{acc}.

10.8.3 Specialized Host Vehicle Tracking Filters

In Kalman filtering, dynamic models are completely specified by two matrix parameters:

1. the dynamic coefficient matrix **F** (or equivalent state transition matrix **Φ**)
2. the dynamic disturbance covariance matrix **Q**.

The values of these matrix parameters for six different vehicle dynamic models are listed in Table 10.5. They are all time invariant (i.e., constant). As a consequence, the corresponding state transition matrices

$$\mathbf{\Phi} = \exp(\mathbf{F}\Delta t)$$

are also constant and can be computed using the matrix exponential function (`expm` in MATLAB®).

Also given in the table is a list of the independent and dependent parameters of the models. The independent parameters can be specified by the filter designer. Because the system model is time invariant, the finite dependent

TABLE 10.5. Vehicle Dynamic Models for GNSS Receivers

Model No.	Model Name	Model Parameters (Each Axis)		Independent Variable	Dependent Variable
		F	**Q**		
1	Unknown constant	0	0	$\sigma^2_{pos}(0)$	None
2	Damped mass–spring system	$\begin{bmatrix} 0 & 1 \\ -\omega^2 - \frac{1}{\tau^2} & \frac{-2}{\tau} \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & \sigma^2_{acc}\Delta t^2 \end{bmatrix}$	σ^2_{pos} ω τ	σ^2_{acc}
3	Type 2 tracker	$\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & \sigma^2_{acc}\Delta t^2 \end{bmatrix}$	σ^2_{acc}	$\sigma^2_{pos} \to \infty$ $\sigma^2_{vel} \to \infty$
4	DAMP1 velocity damping	$\begin{bmatrix} 0 & 1 \\ 0 & -1/\tau_{vel} \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & \sigma^2_{acc}\Delta t^2 \end{bmatrix}$	σ^2_{vel} τ_{vel}	$\sigma^2_{pos} \to \infty$ σ^2_{acc}
5	DAMP2 vel. and acc. damping	$\begin{bmatrix} 0 & 1 & 0 \\ 0 & \frac{-1}{\tau_{vel}} & 1 \\ 0 & 0 & \frac{-1}{\tau_{acc}} \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{jerk}\Delta t^2 \end{bmatrix}$	σ^2_{vel} σ^2_{acc} τ_{acc}	$\sigma^2_{pos} \to \infty$ τ_{vel} $\rho_{vel,acc}$ σ^2_{jerk}
6	DAMP3 pos., vel. and accel. damping	$\begin{bmatrix} \frac{-1}{\tau_{pos}} & 1 & 0 \\ 0 & \frac{-1}{\tau_{vel}} & 1 \\ 0 & 0 & \frac{-1}{\tau_{acc}} \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{jerk}\Delta t^2 \end{bmatrix}$	σ^2_{pos} σ^2_{vel} σ^2_{acc} τ_{acc}	τ_{pos} τ_{vel} $\rho_{pos,vel}$ $\rho_{pos,acc}$ $\rho_{vel,acc}$ σ^2_{jerk}

variables are determinable from the steady-state matrix Riccati differential equation,

$$0 = \mathbf{F}\mathbf{P}_{\infty} + \mathbf{P}_{\infty}\mathbf{F}^{\mathrm{T}} + \mathbf{Q}, \tag{10.132}$$

the solution of which exists only if the eigenvalues of **F** lie in the left-half complex plane. However, even in those cases where the full matrix Riccati differential equation has no finite solution, a reduced equation with a submatrix of $\mathbf{P}_{\infty}$ and corresponding submatrix of **F** may still have a finite steady-state solution. For those with "closed-form" solutions that can be expressed as formulas, the solutions are listed below with the model descriptions.

The TYPE2 filter, for example, does not have a steady-state solution for its Riccati equation without measurements. As a consequence, we cannot use mean-squared velocity as a TYPE2 filter parameter for modeling vehicle maneuverability. However, we can still solve the Riccati equation with GNSS measurements (which is not time invariant) to characterize position uncertainty as a function of mean-squared vehicle acceleration (modeled as a zero-mean white-noise process).

10.8.3.1 Unknown Constant Tracking Model In this model, there are no parameters for vehicle dynamics because there are no vehicle dynamics. The Kalman filter state variables are three components of position, shown below as north–east–down (NED) coordinates. The only model parameters are three values of $\sigma^2_{\mathrm{pos}}(0)$ for three direction components. These represent the initial position uncertainties before measurements start. The value of $\sigma^2_{\mathrm{pos}}(0)$ can be different in different directions. The necessary Kalman filter parameters for a stationary antenna are then

$$\mathbf{P}_0 = \begin{bmatrix} \sigma^2_{\mathrm{north}} & 0 & 0 \\ 0 & \sigma^2_{\mathrm{east}} & 0 \\ 0 & 0 & \sigma^2_{\mathrm{down}} \end{bmatrix}, \text{ the initial mean-squared position uncertainty,}$$

$\mathbf{\Phi} = \mathbf{I}$, the 3×3 identity matrix, and

$\mathbf{Q} = 0$, the 3×3 zero matrix.

The initial position uncertainty, as modeled by $\sigma^2_{\mathrm{pos}}(0)$, may also influence GNSS signal acquisition search time. The other necessary Kalman filter parameters (**H** and **R**) come from the pseudorange measurement model, addressed earlier.

10.8.3.2 Damped Harmonic Resonator GNSS antennas can experience harmonic displacements in the order of several centimeters from host vehicle resonant modes, which are typically at frequencies in the order of ≈1 Hz (the

suspension resonance of most passenger cars) to several hertz—but the effect is small compared to other error sources.

However, a model of this sort (developed in Examples 10.1–10.8) is needed for INS gyrocompass alignment, which is addressed in Chapter 11.

10.8.3.3 Type 2 Tracking Model The TYPE2 tracker is older than Kalman filtering. Given sufficient measurements, it can estimate position and velocity in three dimensions. (Type 1 trackers do not estimate velocity.) The tracker uses a host vehicle dynamic model with zero-mean white-noise acceleration, unbounded steady-state mean-squared velocity (not particularly reasonable) and unbounded steady-state mean-squared position variation (quite reasonable). When GNSS signals are lost, the velocity uncertainty variance will grow without bound unless something is done about it—such as limiting velocity variance to some maximum value. Trackers based on this model can do an adequate job when GNSS signals are present.

The model parameters shown in Table 10.5 are for a single-direction component and do not include position. The full tracking model will include three position components and three velocity components. The necessary Kalman filter parameters for a 3D Type 2 tracking filter include

$$\mathbf{P}_0 = \begin{bmatrix} \sigma^2_{\text{north}} & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma^2_{\text{east}} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma^2_{\text{down}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^2_{v,\text{north}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma^2_{v,\text{east}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma^2_{v,\text{down}} \end{bmatrix},$$

$$\mathbf{\Phi} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^2_{\text{acc}}\Delta t^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma^2_{\text{acc}}\Delta t^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma^2_{\text{acc}}\Delta t^2 \end{bmatrix},$$

where σ^2_{acc} is the only adjustable parameter value. Adjusting it for a particular application may take some experimenting.

10.8.3.4 DAMP1 Tracking Model: Velocity Damping This type of tracking filter is based on the Langevin equation,

$$\frac{d}{dt}v(t) = \underbrace{-\frac{1}{\tau_{\text{vel}}}}_{F} v(t) + w(t), \tag{10.133}$$

where $v(t)$ is a velocity component and $w(t)$ is a zero-mean white-noise process in continuous time.

It differs from the TYPE2 tracker in that it includes a velocity damping time constant τ_{vel}, which is enough to put an eigenvalue of **F** in the left-half complex plane and allow a steady-state variance for velocity. This is more realistic as a model for a vehicle with finite speed capabilities. Also, the parameter τ_{vel} is a measure of persistence of velocity, which would be useful for distinguishing the dynamics of an oil tanker, say, from those of a jet ski.

The values of $\mathbf{P}_0$ and **Q** will be the same as for the TYPE2 tracker, and the state transition matrix

$$\mathbf{\Phi} = \begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t \\ 0 & 0 & 0 & \varepsilon & 0 & 0 \\ 0 & 0 & 0 & 0 & \varepsilon & 0 \\ 0 & 0 & 0 & 0 & 0 & \varepsilon \end{bmatrix}$$

$$\varepsilon = \exp(-\Delta t / \tau_{\text{vel}}).$$

The steady-state solution of the Riccati equation can be used to solve for

$$\sigma^2_{\text{acc}} = \sigma^2_{\text{vel}}[1 - \exp(-2\Delta t / \tau_{\text{vel}})] / \Delta t^2; \tag{10.134}$$

that is, one can specify the vehicle maneuver capability in terms of its mean-square velocity σ^2_{vel} and velocity correlation time τ_{vel}, and use Eq. 10.134 to specify compatible values for the modeled **Q** matrix.

10.8.3.5 DAMP2 Tracking Model: Velocity and Acceleration Damping This is an even more realistic model for a vehicle with finite speed and acceleration capabilities. It also includes an acceleration time-correlation constant τ_{acc}, which is useful from distinguishing the more lively vehicle types from the more sluggish ones. The corresponding steady-state Riccati equation used for

making the model a function of RMS velocity and acceleration is not as easy to solve in closed form, however.

The 2×2 submatrix of the state transition matrix $\mathbf{\Phi}$ relating velocity and acceleration along a single axis has the form

$$\mathbf{\Phi}_{\text{vel,acc}} = \begin{bmatrix} \phi_{1,1} & \phi_{1,2} \\ \phi_{2,1} & \phi_{2,2} \end{bmatrix} \tag{10.135}$$

$$\phi_{1,1} = \exp(-\Delta t / \tau_{\text{vel}}) \tag{10.136}$$

$$\phi_{1,2} = \frac{\tau_{\text{vel}} \tau_{\text{acc}} [\exp(-\Delta t / \tau_{\text{vel}}) - \exp(-\Delta t / \tau_{\text{acc}})]}{\tau_{\text{vel}} - \tau_{\text{acc}}} \tag{10.137}$$

$$\phi_{2,1} = 0 \tag{10.138}$$

$$\phi_{2,2} = \exp(-\Delta t / \tau_{\text{acc}}), \tag{10.139}$$

and the corresponding 2×2 submatrix of $\mathbf{Q}$ will be

$$\mathbf{Q}_{\text{vel,acc}} = \begin{bmatrix} 0 & 0 \\ 0 & \sigma_{\text{jerk}}^2 \Delta t^2 \end{bmatrix}. \tag{10.140}$$

The corresponding steady-state Riccati Eq. 10.132 can be solved for

$$\sigma_{\text{jerk}}^2 = \sigma_{\text{acc}}^2 [1 - \exp(-2\Delta t / \tau_{\text{acc}})] / \Delta t^2, \tag{10.141}$$

the analog of Eq. 10.134.

The steady-state Riccati equation can also be solved for the correlation coefficient

$$\rho_{\text{vel,acc}} = -\frac{\tau_{\text{vel}} \tau_{\text{acc}} \sigma_{\text{acc}} \exp(-\Delta t / \tau_{\text{acc}}) [\exp(-\Delta t / \tau_{\text{vel}}) - \exp(-\Delta t / \tau_{\text{acc}})]}{\sigma_{\text{vel}} [1 - \exp(-\Delta t / \tau_{\text{acc}}) \exp(-\Delta t / \tau_{\text{vel}})] (\tau_{\text{vel}} - \tau_{\text{acc}})} \tag{10.142}$$

between the velocity and acceleration components. But solving the remaining element $p_{\infty,1,1} = 0$ of the Riccati Eq. 10.132 for τ_{vel} as variable dependent on the independent variables requires solving a transcendental equation. It is solved numerically in the MATLAB® function `Damp2Params.m` on the accompanying website.

Figure 10.15 was generated by the MATLAB® m-file `Damp2eval.m`, and Fig. 10.17 was generated by the m-file `SchmidtKalmanTest.m` on the accompanying website. Both include solutions of the Riccati equation for a DAMP2 GNSS position filter.

10.8.3.6 DAMP3 Tracking Model: Position, Velocity, and Acceleration Damping This type of filter is designed for vehicles with limited but nonzero position variation, such as the altitudes of some surface watercraft (e.g., river boats) and land vehicles. Ships that remain at sea level are at zero altitude, by definition. They need no vertical navigation, unless they are trying to estimate tides. Flatwater boats and land vehicles in very flat areas can probably do without vertical navigation, as well.

Continuous-Time Solutions It is generally easier to solve the steady-state covariance equation in continuous time,

$$0 = \mathbf{F}_3\mathbf{P}_3 + \mathbf{P}_3\mathbf{F}_3^{\mathrm{T}} + \mathbf{Q}_3, \tag{10.143}$$

for the parameters in the steady-state solution

$$\mathbf{P}_3 = \begin{bmatrix} p_{1,1} & p_{1,2} & p_{1,3} \\ p_{1,2} & p_{2,2} & p_{2,3} \\ p_{1,3} & p_{2,3} & p_{3,3} \end{bmatrix}, \tag{10.144}$$

where the other model parameters are

$$\mathbf{F}_3 = \begin{bmatrix} -\tau_{\mathrm{pos}}^{-1} & 1 & 0 \\ 0 & -\tau_{\mathrm{vel}}^{-1} & 1 \\ 0 & 0 & -\tau_{\mathrm{acc}}^{-1} \end{bmatrix} \tag{10.145}$$

$$\mathbf{Q}_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & q_{c,3,3} \end{bmatrix}. \tag{10.146}$$

The six scalar equations equivalent to the symmetric 3×3 matrix Eq. 10.143 are

$$\left.\begin{aligned} 0 &= -p_{1,1} + p_{1,2}\tau_{\mathrm{pos}} && (\mathrm{Eq}_{1,1}) \\ 0 &= -p_{1,2}\tau_{\mathrm{vel}} + p_{2,2}\tau_{\mathrm{pos}}\tau_{\mathrm{vel}} - p_{1,2}\tau_{\mathrm{pos}} + p_{1,3}\tau_{\mathrm{pos}}\tau_{\mathrm{vel}} && (\mathrm{Eq}_{1,2}) \\ 0 &= -p_{1,3}\tau_{\mathrm{acc}} + p_{2,3}\tau_{\mathrm{pos}}\tau_{\mathrm{acc}} - p_{1,3}\tau_{\mathrm{pos}} && (\mathrm{Eq}_{1,3}) \\ 0 &= -p_{2,2} + p_{2,3}\tau_{\mathrm{vel}} && (\mathrm{Eq}_{2,2}) \\ 0 &= -p_{2,3}\tau_{\mathrm{acc}} + p_{3,3}\tau_{\mathrm{vel}}\tau_{\mathrm{acc}} - p_{2,3}\tau_{\mathrm{vel}} && (\mathrm{Eq}_{2,3}) \\ 0 &= -2p_{3,3} + q_{c,3,3}\tau_{\mathrm{acc}} && (\mathrm{Eq}_{3,3}) \end{aligned}\right\}. \tag{10.147}$$

We wish to solve for the steady-state covariance matrix **P**, where the independent variables

τ_{acc}, the acceleration correlation time constant,
$p_{1,1}$, the mean-squared position excursion,
$p_{2,2}$, the mean-squared velocity variation, and
$p_{3,3}$, the mean-squared acceleration variation

are to be specified, and the dependent variables

τ_{vel}, the velocity correlation time constant,
τ_{pos}, the position correlation time constant,
$p_{1,2}$, the cross covariance of position and velocity,
$p_{1,3}$, the cross covariance of position and acceleration,
$p_{2,3}$, the cross covariance of velocity and acceleration, and
$q_{c,3,3}$, the continuous-time disturbance noise variance

are to be determined from Eq. 10.147.

From the last of these (labeled $\text{Eq}_{3,3}$),

$$q_{c,3,3} = 2\frac{p_{3,3}}{\tau_{acc}}. \tag{10.148}$$

From $\text{Eq}_{2,2}$ and $\text{Eq}_{2,3}$,

$$p_{2,3} = \frac{p_{2,2}}{\tau_{vel}} \tag{10.149}$$

$$= \frac{p_{3,3}\tau_{vel}\tau_{acc}}{\tau_{acc} + \tau_{vel}}, \tag{10.150}$$

and from equating the two solutions,

$$\tau_{vel} = \frac{p_{2,2} + \sqrt{p_{2,2}^2 + 4p_{3,3}\tau_{acc}^2 p_{2,2}}}{2p_{3,3}\tau_{acc}}. \tag{10.151}$$

Similarly, from $\text{Eq}_{1,1}$ and $\text{Eq}_{1,2}$,

$$p_{1,2} = \frac{p_{1,1}}{\tau_{pos}} \tag{10.152}$$

$$p_{1,3} = -\frac{p_{2,2}\tau_{pos}^2\tau_{vel} - p_{1,1}\tau_{vel} - p_{1,1}\tau_{pos}}{\tau_{pos}^2\tau_{vel}}, \tag{10.153}$$

and from $\mathrm{Eq}_{1,3}$,

$$p_{1,3} = \frac{p_{2,3}\tau_{\mathrm{pos}}\tau_{\mathrm{acc}}}{\tau_{\mathrm{acc}} + \tau_{\mathrm{pos}}}. \tag{10.154}$$

By equating the two independent solutions for $p_{1,3}$, one obtains a cubic polynomial in τ_{pos}:

$$0 = c_0 + c_1\tau_{\mathrm{pos}} + c_2\tau_{\mathrm{pos}}^2 + c_3\tau_{\mathrm{pos}}^3 \tag{10.155}$$

with coefficients

$$c_0 = p_{1,1}\tau_{\mathrm{vel}}\tau_{\mathrm{acc}} \tag{10.156}$$

$$c_1 = p_{1,1}(\tau_{\mathrm{acc}} + \tau_{\mathrm{vel}}) \tag{10.157}$$

$$c_2 = -p_{2,2}\tau_{\mathrm{vel}}\tau_{\mathrm{acc}} + p_{1,1} \tag{10.158}$$

$$c_3 = -\tau_{\mathrm{vel}}(p_{2,3}\tau_{\mathrm{acc}} + p_{2,2}), \tag{10.159}$$

which can be solved numerically using the MATLAB® function `roots`.

The MATLAB solution sequence is then

0. Given: $p_{1,1}$, $p_{2,2}$, $p_{3,3}$, and τ_{acc}.
1. Solve for τ_{vel} using Eq. 10.151.
2. Solve for $p_{2,3}$ using Eq. 10.149.
3. Solve for τ_{pos} using Eq. 10.155 and the MATLAB® function roots.
4. Solve for $p_{1,2}$ using Eq. 10.152.
5. Solve for $p_{1,3}$ using Eq. 10.154.

This solution is implemented in the MATLAB® m-file `DAMP3Params.m` on the accompanying website.

This leaves the problem of solving for the discrete-time process noise covariance matrix $\mathbf{Q}_{3,\mathrm{discrete}}$, which is not the same as the analogous matrix $\mathbf{Q}_3$ in continuous time (solved using Eq. 10.148). There is a solution formula,

$$\mathbf{Q}_{3,\mathrm{discrete}} = \exp(\Delta t\mathbf{F})\left[\int_0^{\Delta t}\exp(-s\mathbf{F})\mathbf{Q}_3\exp(-s\mathbf{F}^{\mathrm{T}})\,ds\right]\exp(\Delta t\mathbf{F}^{\mathrm{T}}), \tag{10.160}$$

but—given $\boldsymbol{\Phi}$ and $\mathbf{P}_\infty$—it is easier to use the steady-state formula

$$\mathbf{Q}_{3,\mathrm{discrete}} = \mathbf{P}_3 - \boldsymbol{\Phi}\mathbf{P}_3\boldsymbol{\Phi}^{\mathrm{T}}, \tag{10.161}$$

which is how the solution is implemented in `DAMP3Params.m`.

10.8.3.7 Tracking Models for Highly Constrained Trajectories Race cars in some televised races have begun using integrated GNSS/INS on each vehicle to determine their positions on the track. The estimated positions are telemetered to the television control system, where they are used in generating television graphics (e.g., an arrow or dagger icon) to designate on the video images where each car is on the track at all times. The integrating filters constrain the cars to be on the 2D track surface, which improves the estimation accuracy considerably.

FIG8 Tracking Model As a simple example of how this works, we will use a one-dimensional "slot car" track model, with the position on the track completely specified by the down-track distance from a reference point.

The trajectory of a vehicle on the track is specified in terms of a formula,

$$\boldsymbol{\delta}_{\text{pos}} = \begin{bmatrix} \text{Northing} \\ \text{Easting} \\ -\text{Altitude} \end{bmatrix} \tag{10.162}$$

$$= \begin{bmatrix} 3S\sin(\omega t + \phi) \\ 2S\sin(\omega t + \phi)\cos(\omega t + \phi) \\ -1/2h\cos(\omega t + \phi) \end{bmatrix}, \tag{10.163}$$

where

S is a track scaling parameter, ≈ (track length [m])/14.9437552901562;
h is half the vertical separation where the track crosses over itself;
$\omega = 2\pi \times$ (average speed [m/s])/(track length [m]); and
ϕ is an arbitrary phase angle (rad).

The phase rate $\dot{\phi}$ can be modeled as a random walk or exponentially correlated process, to simulate speed variations. This model is implemented in the MATLAB® function `Fig8Mod1D`, which also calculates vehicle velocity, acceleration, attitude and attitude rates. This m-file is on the accompanying website. The resulting trajectory is illustrated in Fig. 10.16.

The resulting Kalman filter is implemented in the MATLAB® m-file `GPSTrackingDemo.m` on the accompanying website. This particular implementation is for a 1.5-km track with vehicle speeds of 90 kph ± 10% RMS random variation. The Kalman filter model in `GPSTrackingDemo.m` uses only two vehicle states: (1) the phase angle ϕ and (2) its derivative $\dot{\phi}$ which is modeled as an exponentially correlated random process with a correlation time constant of 10 s and RMS value equivalent to ±10% variation in speed.

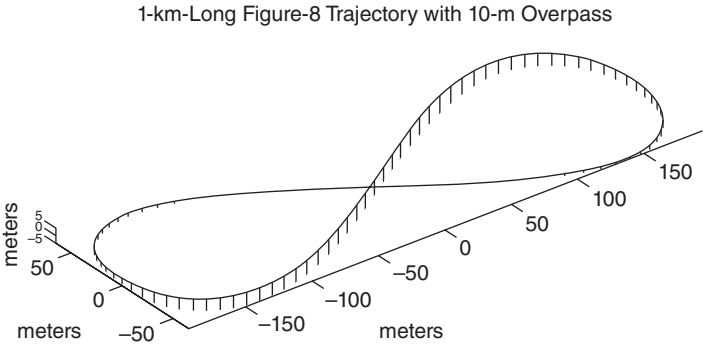


Fig. 10.16 Figure-8 trajectory of length 1500 m.

10.8.3.8 Filters for Spacecraft Unpowered vehicles in space do not have sufficiently random dynamics to justify a tracking filter. They may have unknown, quasi-constant orbit parameters, but their trajectories over the short-term are essentially defined by a finite set of parameters. GNSS vehicle tracking then becomes an orbit determination problem. The orbital parameters may change during brief orbit changes, but the problem remains an orbit determination problem with increased uncertainty in initial conditions (velocity, in particular).

10.8.3.9 Other Specialized Vehicle Filter Models The list of models in Table 10.5 is by no means exhaustive. It does not include the FIG8 filter described above. Other specialized filters have been designed for vehicles confined to narrow corridors within a limited area, such as race cars on a 2D track or motor vehicles on streets and highways. Specialized filters are also required for trains, which need to know where they are on a 1D track, and possibly which set of parallel rails they are on.

Still, the models listed in Table 10.5 and described above should cover the majority of GNSS applications.

10.8.3.10 Filters for Different Host Vehicle Types Table 10.6 lists some generic host vehicle types, along with names of models in Table 10.5 that might be used for GNSS position tracking on such vehicles.

10.8.3.11 Parameters for Vehicle Dynamics Table 10.7 contains descriptions of the tracking filter parameters shown in Table 10.5. These are statistical parameters for characterizing random dynamics of the host vehicle.

10.8.3.12 Empirical Modeling of Vehicle Dynamics The most reliable vehicle dynamic models are those based on data from representative vehicle dynamics. Empirical modeling of the uncertain dynamics of host vehicles

TABLE 10.6. Filter Models for Unknown Vehicle Dynamics

	Filter Models	
Host Vehicles	Horizontal Directions	Vertical Direction
None (fixed to earth)	Unknown constant	Unknown constant
Parked	Damp. harm. resonator	Damp. harm. resonator
Ships	DAMP1, DAMP2	Unknown constant
Land vehicles	DAMP1, DAMP2	DAMP3
Aircraft and missiles	DAMP1, DAMP2	DAMP2, DAMP3
Spacecraft	In free-fall: Use orbit estimation models After maneuvers: Increment velocity uncertainty	

TABLE 10.7. Statistical Parameters of Host Vehicle Dynamics

Symbol	Definition
σ^2_{pos}	Mean-squared position excursions[a]
σ^2_{vel}	Mean-squared vehicle velocity ($E\langle\|\mathbf{v}\|^2\rangle$)
σ^2_{acc}	Mean-squared vehicle acceleration ($E\langle\|\mathbf{a}\|^2\rangle$)
σ^2_{jerk}	Mean-squared jerk ($E\left\langle\|\dot{\mathbf{a}}\|^2\right\rangle$)
$\rho_{i,j}$	Correlation coefficients between position, velocity, and acceleration variations
τ_{pos}	Position correlation time
τ_{vel}	Velocity correlation time
τ_{acc}	Acceleration correlation time
ω_{resonant}	Suspension resonant frequency
τ_{damping}	Suspension damping time constant

[a]Mean-squared position excursions generally grow without bound, except for altitudes of ships (and possibly land vehicles).

requires data (i.e., position and attitude and their derivatives) recorded under conditions representing the intended mission applications.

The ideal sensor for this purpose is an INS, or at least an inertial sensor assembly (ISA) capable of measuring and recording 3D accelerations and attitude rates (or attitudes) during maneuvers of the host vehicle.

The resulting data are the sum of three types of motion:

1. Internal motions due to vibrating modes of the vehicle, excited by propulsion noise and flow noise in the surrounding medium. The oscillation periods for this noise generally scale with the size of the host vehicle but are generally in the order of a second or less.
2. Short-term perturbations of the host vehicle that are corrected by steering, such as turbulence acting on aircraft or potholes acting on wheeled vehicles. These also excite the vibrational modes of the vehicle.

3. The intended rigid-body motions of the whole vehicle to follow the planned trajectory. The frequency range of these motions is generally 10 Hz and often <1 Hz.

Only the last of these is of interest in tracking. It can often be separated from the high-frequency noise by low-pass filtering, ignoring the high-frequency end of the power spectral densities and cross-spectral densities of the data. The inverse Fourier transforms of the low-end power spectral data will yield autocovariance functions that are useful for modeling purposes. The statistics of interest in these autocovariance functions are the variances σ^2 (values at zero correlation time) and the approximate exponential decay times τ of the autocovariances.

10.8.4 Vehicle Tracking Filter Comparison

The alternative GNSS receiver tracking filters of the previous section were evaluated using the "figure-eight" track model described in Section 10.8.3. This is a trajectory confined in all three dimensions, and more in some dimensions than others.

10.8.4.1 Simulated Trajectory The simulated trajectory is that of an automobile on a banked figure-eight track, as illustrated in Fig. 10.16. The MATLAB® m-file `Fig8TrackDemo.m` on the accompanying website generates a series of plots and statistics of the simulated trajectory. It calls the MATLAB® function `Fig8Mod1D`, which generates the simulated dynamic conditions on the track, and it outputs the following statistics of nominal dynamic conditions:

```
RMS N-S Position Excursion = 212.9304 meter
RMS E-W Position Excursion = 70.9768 meter
RMS Vert. Position Excursion = 3.5361 meter
RMS N-S Velocity = 22.3017 m/s
RMS E-W Velocity = 14.8678 m/s
RMS Vert. Velocity = 0.37024 m/s
RMS N-S Acceleration = 2.335 m/s/s
RMS E-W Acceleration = 3.1134 m/s/s
RMS Vert. Acceleration = 0.038778 m/s/s
RMS Delta Velocity North = 0.02335 m/s at Δt = 0.01 sec.
                         = 2.334 m/s at Δt = 1 sec.
RMS Delta Velocity East = 0.031134 m/s at Δt = 0.01 sec.
                        = 3.1077 m/s at Δt = 1 sec.
RMS Delta Velocity Down = 0.00038771 m/s at Δt = 0.01 sec.
                        = 0.038754 m/s at Δt = 1 sec.
```

```
N. Position Correlation Time = 13.4097 sec.
E. Position Correlation Time = 7.6696 sec.
Vertical Position Corr. Time = 9.6786 sec.
N. Velocity Correlation Time = 9.6786 sec.
E. Velocity Correlation Time = 21.4921 sec.
Vertical Velocity Corr. Time = 13.4097 sec.
N. Acceler. Correlation Time = 13.4097 sec.
E. Acceler. Correlation Time = 7.6696 sec.
Vertical Acceler. Corr. Time = 9.6786 sec.
```

These statistics are used for "tuning" the filter parameters for each of the alternative vehicle tracking filters—within the capabilities of the tracking filter.

10.8.4.2 Results The MATLAB® m-file `GPSTrackingDemo.m` on the accompanying website simulates the GNSS satellites, the vehicle, and all four types of filters on a common set of pseudorange measurements over a period of two hours. The position estimation results are summarized in Table 10.8 for one particular simulation. Even though all models were "optimized" using the same statistical parameters for the vehicle trajectory, depending on which vehicle model is chosen, the results can differ by two orders of magnitude in RMS position error.

The m-file `GPSTrackingDemo.m` generates many more plots to demonstrate how the different filters are working, including plots of the simulated and estimated pseudorange errors for each of the 29 satellites, most of which are not in view. Because the simulation uses a pseudorandom number generator, the results can change from run to run.

10.8.4.3 Model Dimension versus Model Constraints These results indicate that dilution of information is not just a matter of state vector dimension. One might expect that the more variables there are to estimate, the less information will be available for each variable. In Table 10.8, the DAMP3 model

TABLE 10.8. Comparison of Alternative GNSS Filters on 1.5-km Figure-8 Track Simulation

	RMS Pos. Est. Err.[a] (m)		
GNSS Filter	North	East	Down
TYPE2	42.09	40.71	4.84
DAMP2	22.98	25.00	3.51
DAMP3	7.34	10.52	3.31
FIG8	0.53	0.31	0.01

[a]Clock errors not included.

has three more state variables than the TYPE2 or DAMP2 models, yet it produces better results. The other issue at work here is the degree to which the model constrains the solution, and this factor better explains the ordering of estimation accuracy. The degree to which the model constrains the solution increases downward in the table, and simulated performance improves monotonically with the degree of constraint.

10.8.4.4 Role of Model Fidelity These results strongly suggest some performance advantage to be gained by tuning the vehicle tracking filter structure and parameters to the problem at hand. For the simulated trajectory, the accelerations, velocities, and position excursions are all constrained, and the model that takes greatest advantage of that is FIG8, the track-specific model.

10.9 ALTERNATIVE IMPLEMENTATIONS

There have been many "improvements" in the Kalman filter since 1960. Some are changes in the methods of computation, some use the Kalman filter model to solve related nonfiltering problems, and some make use of Kalman filtering variables for addressing other application-related problems. We present here some that have been found useful in GNSS/INS integration. More extensive coverage of the underlying issues and solution methods is provided in Ref. 12.

10.9.1 Schmidt–Kalman Suboptimal Filtering

This is a method proposed by Stanley F. Schmidt [26] for reducing the processing and memory requirements for Kalman filtering, with predictable performance degradation. At the time it was proposed (in the mid-1960s), computer capabilities were severely limited compared to those available today, and much effort had to be put into reducing computation and memory requirements as much as possible. Programmers needed to be stingy with computation cycles and bits.

Schmidt–Kalman (SK) filtering has been used as a means of eliminating the additional variables (one per GNSS satellite used) required for Kalman filtering with time-correlated pseudorange errors (originally for SA errors, but also useful for uncompensated ionospheric propagation delays). However, before taking that approach, potential users should also calculate the actual savings in memory and processor time before deciding whether it is worth the performance degradation and the added risk of programming a much more complex implementation.

The algorithms are presented here because they have been used in some GNSS receiver implementations, but they are not recommended.

10.9.1.1 State Vector Partitioning SK filtering partitions the state vector into "essential" variables (designated by the subscript e) and "unessential" variables (designated by the subscript u):

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_{\mathrm{e}} \\ \mathbf{x}_{\mathrm{u}} \end{bmatrix}, \tag{10.164}$$

where $\mathbf{x}_{\mathrm{e}}$ is the $n_{\mathrm{e}} \times 1$ subvector of essential variables to be estimated, $\mathbf{x}_{\mathrm{u}}$ is the $n_{\mathrm{u}} \times 1$ subvector that will not be estimated, and

$$n_{\mathrm{e}} + n_{\mathrm{u}} = n, \text{ the total number of state variables.} \tag{10.165}$$

Even though the subvector $\mathbf{x}_{\mathrm{u}}$ of nuisance variables is not estimated, the effects of not doing so must be reflected in the covariance matrix $\mathbf{P}_{\mathrm{ee}}$ of uncertainty in the estimated variables. For that purpose, the SK filter calculates the covariance matrix $\mathbf{P}_{\mathrm{uu}}$ of uncertainty in the unestimated state variables and the cross-covariance matrix $\mathbf{P}_{\mathrm{ue}}$ between the two types. These other covariance matrices are used in the calculation of the SK gain.

10.9.1.2 Implementation Equations The essential implementation equations for the SK filter are listed in Table 10.9. These equations have

TABLE 10.9. Summary Implementation of Schmidt–Kalman Filter

Corrector (Observational Update)		
$\mathbf{C}$	=	$\mathbf{P}_{ee,k}(-)\mathbf{H}_{e,k}^{\mathrm{T}} + \mathbf{P}_{eu,k}(-)\mathbf{H}_{u,k}^{\mathrm{T}}$
$\mathbf{D}$	=	$\mathbf{P}_{ue,k}(-)\mathbf{H}_{e,k}^{\mathrm{T}} + \mathbf{P}_{uu,k}(-)\mathbf{H}_{u,k}^{\mathrm{T}}$
$\mathbf{E}$	=	$\mathbf{H}_{e,k}\mathbf{C} + \mathbf{H}_{u,k}\mathbf{D} + \mathbf{R}_k$
$\bar{\mathbf{K}}_{\mathrm{SK},k}$	=	$\mathbf{C}\mathbf{E}^{-1}$
$\hat{\mathbf{x}}_{e,k}(+)$	=	$\hat{\mathbf{x}}_{e,k}(-) + \bar{\mathbf{K}}_{\mathrm{SK},k}[\mathbf{z}_k - \mathbf{H}_{ek}\mathbf{x}_{e,k}(-)]$
$\mathbf{A}$	=	$\mathbf{I}_{n_e} - \bar{\mathbf{K}}_{\mathrm{SK},k}\mathbf{H}_{e,k}$
$\mathbf{B}$	=	$\bar{\mathbf{K}}_{\mathrm{SK},k}\mathbf{H}_{u,k}$
$\mathbf{P}_{eu,k}(+)$	=	$\mathbf{A}\mathbf{P}_{eu,k}(-) - \mathbf{B}\mathbf{P}_{uu,k}(-)$
$\mathbf{P}_{ee,k}(+)$	=	$[\mathbf{A}\mathbf{P}_{ee,k}(-) - \mathbf{B}\mathbf{P}_{eu,k}(-)^T]\mathbf{A}^T$ $-\mathbf{P}_{eu,k}(-)\mathbf{B}^T + \mathbf{K}_{\mathrm{SK},k}\mathbf{R}_k\mathbf{K}_{\mathrm{SK},k}^T$
$\mathbf{P}_{ue,k}(+)$	=	$\mathbf{P}_{eu,k}(+)^{\mathrm{T}}$
$\mathbf{P}_{uu,k}(+)$	=	$\mathbf{P}_{uu,k}(-)$
Predictor (Time Update)		
$\hat{\mathbf{x}}_{e,k+1}(-)$	=	$\mathbf{\Phi}_{e,k}\hat{\mathbf{x}}_{e,k}(+)$
$\mathbf{P}_{ee,k+1-}$	=	$\mathbf{\Phi}_{e,k}\mathbf{P}_{ee,k}(+)\mathbf{\Phi}_{e,k}^{\mathrm{T}} + \mathbf{Q}_{ee}$
$\mathbf{P}_{eu,k+1}(-)$	=	$\mathbf{\Phi}_{e,k}\mathbf{P}_{eu,k}(+)\mathbf{\Phi}_{u,k}^{\mathrm{T}}$
$\mathbf{P}_{ue,k+1}(-)$	=	$\mathbf{P}_{eu,k+1}(-)^{\mathrm{T}}$
$\mathbf{P}_{uu,k+1-}$	=	$\mathbf{\Phi}_{u,k}\mathbf{P}_{uu,k+}\mathbf{\Phi}_{u,k}^{\mathrm{T}} + \mathbf{Q}_{uu}$

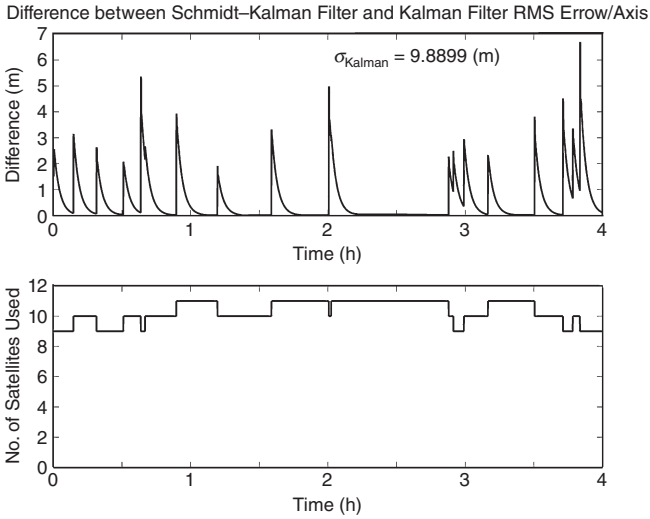


Fig. 10.17 Simulation comparing Schmidt–Kalman and Kalman filters.

been arranged for reusing intermediate results to reduce computational requirements.

10.9.1.3 Simulated Performance in GNSS Position Estimation Figure 10.17 is the output of the MATLAB® m-file `SchmidtKalmanTest.m` on the accompanying website. This is a simulation using the vehicle dynamic model DAMP2, described in Section 10.8.3, with 29 GNSS satellites (almanac of 08 Mar 2006), 9–11 of which were in view (15° above the horizon) at any one time, and

$n_e = 9$, the number of essential state variables (3 ea. of pos., vel., acc.)

$n_u = 29$, the number of unessential state variables (propagation delays)

$\Delta t = 1$ s, time interval between filter updates

$\sigma_{\text{pos}}(0) = 20$ m, initial position uncertainty, RMS/axis

$\sigma_{\text{vel}} = 200$ m/s, RMS of random vehicle speed (about 447 mph)

$\sigma_{\text{acc}} = 1/2$ g, RMS of random vehicle acceleration

$\sigma_{\text{prop}} = 10$ m, RMS propagation delay uncertainty (steady-state)

$\sigma_\rho = 10$ m, RMS pseudorange measurement noise (white, zero-mean)

$\tau_{\text{prop}} = 150$ s, correlation time constant of propagation delay

$\tau_{\text{acc}} = 120$ s, correlation time constant of random vehicle acceleration.

The variables plotted in the top graph are the "un-RSS" differences

$$\sigma_{\text{difference}} = \sqrt{\sigma_{\text{SK}}^2 - \sigma_{\text{K}}^2}$$

between the mean-squared position uncertainties of the SK filter (σ_{SK}^2) and the Kalman filter ($\sigma_{\text{K}}^2 \approx 10^2$ m^2). The peak errors introduced by the SK filter are a few meters to several meters and are transient. The error spikes generally coincide with the changes in the number of satellites used (plotted in the bottom graph). This would indicated that, for this GNSS application anyway, the SK filter performance comes very close to that of the Kalman filter, except when a new satellite with unknown propagation delay is first used. Even then, the errors introduced by using a new satellite generally die down after a few correlation time constants of the propagation delay errors.

10.9.2 Serial Measurement Processing

It is shown in Ref. 18 that it is more efficient to process the components of a measurement vector serially, one component at a time, than to process them as a vector. This may seem counterintuitive, but it is true even if its implementation requires a transformation of measurement variables to make the associated measurement noise covariance $\mathbf{R}$ a diagonal matrix (i.e., with noise uncorrelated from one component to another).

10.9.2.1 Measurement Decorrelation If the covariance matrix $\mathbf{R}$ of measurement noise is *not* a diagonal matrix, then it can be made so by $\mathbf{UDU}^{\text{T}}$ decomposition (Eq. B.22) and by changing the measurement variables[10]:

$$\mathbf{R}_{\text{correlated}} = \mathbf{U}_R \mathbf{D}_R \mathbf{U}_R^{\text{T}}, \tag{10.166}$$

$$\mathbf{R}_{\text{decorrelated}} \overset{\text{def}}{=} \mathbf{D}_R \text{ (a diagonal matrix)}, \tag{10.167}$$

$$\mathbf{z}_{\text{decorrelated}} \overset{\text{def}}{=} \mathbf{U}_R \setminus \mathbf{z}_{\text{correlated}}, \tag{10.168}$$

$$\mathbf{H}_{\text{decorrelated}} \overset{\text{def}}{=} \mathbf{U}_R \setminus \mathbf{H}_{\text{correlated}}, \tag{10.169}$$

where $\mathbf{R}_{\text{correlated}}$ is the nondiagonal (i.e., correlated component-to-component) measurement noise covariance matrix, and the new *decorrelated* measurement

[10]We use the matrix divide symbol "\" in place of the matrix inverse on $\mathbf{U}_{\text{R}}$ because it is triangular matrix, and the system can be solved more easily by back-substitution.

TABLE 10.10. Implementation Equations for Serial Measurement Update

$\mathbf{x} = \hat{\mathbf{x}}_k(-)$
$\mathbf{P} = \mathbf{P}_k(-)$
for j=1:ℓ,
$z = \mathbf{z}_k(\mathrm{j})$;
$\mathbf{H} = \mathbf{H}_k(\mathrm{j}, :)$;
$R = \mathbf{R}_{\text{decorrelated}}(\mathrm{j}, \mathrm{j})$;
$\bar{\mathbf{K}} = \mathbf{PH}'/(\mathbf{HPH}' + R)$
$\hat{\mathbf{x}} = \bar{\mathbf{K}}(z - \mathbf{Hx})$
$\mathbf{P} = \mathbf{P} - \bar{\mathbf{K}}\mathbf{HP}$
end;
$\hat{\mathbf{x}}_k(+) = \mathbf{x}$
$\mathbf{P}_k(+) = (\mathbf{P} + \mathbf{P}')/2$; (symmetrize)

vector $\mathbf{z}_{\text{decorrelated}}$ has a diagonal measurement noise covariance matrix $\mathbf{R}_{\text{decorrelated}}$ and measurement sensitivity matrix $\mathbf{H}_{\text{decorrelated}}$.

10.9.2.2 Serial Processing of Decorrelated Measurements The components of $z_{\text{decorrelated}}$ can now be processed one component at a time using the corresponding row of $\mathbf{H}_{\text{decorrelated}}$ as its measurement sensitivity matrix and the corresponding diagonal element of $\mathbf{R}_{\text{decorrelated}}$ as its measurement noise variance.

A "pidgin-MATLAB®" implementation for this procedure is listed in Table 10.10, where the final line is a "symmetrizing" procedure designed to improve robustness.

10.9.3 Improving Numerical Stability

10.9.3.1 Effects of Finite Precision Computer roundoff limits the precision of numerical representation in the implementation of Kalman filters. It has been known to cause severe degradation of filter performance in many applications, and alternative implementations of the Kalman filter equations (the Riccati equations, in particular) have been shown to improve robustness against roundoff errors.

Computer roundoff for floating-point arithmetic is often characterized by a single parameter $\varepsilon_{\text{roundoff}}$, which is the smallest number such that

$$1 + \varepsilon_{\text{roundoff}} > 1 \text{ in machine precision.} \tag{10.170}$$

It is the value assigned to the parameter eps in MATLAB®. In 64-bit ANSI/IEEE Standard floating point arithmetic (MATLAB® precision on PCs), eps = 2^{-52}.

The following example, due to Dyer and McReynolds [9], shows how a problem that is well conditioned, as posed, can be made ill-conditioned by the filter implementation.

Example 10.9 Ill-Conditioned Measurement Sensitivity. Consider the filtering problem with measurement sensitivity matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1+\delta \end{bmatrix}$$

and covariance matrices

$$\mathbf{P}_0 = \mathbf{I}_3, \textit{ and } \mathbf{R} = \delta^2 \mathbf{I}_2,$$

where $\mathbf{I}_n$ denotes the $n \times n$ identity matrix and the parameter δ satisfies the constraints

$$\delta^2 < \varepsilon_{\text{roundoff}}, \text{ but } \delta > \varepsilon_{\text{roundoff}}.$$

In this case, although $\mathbf{H}$ clearly has rank 2 in machine precision, the product $\mathbf{H}\mathbf{P}_0\mathbf{H}^{\mathrm{T}}$ with roundoff will equal

$$\begin{bmatrix} 3 & 3+\delta \\ 3+\delta & 3+2\delta \end{bmatrix},$$

which has a pseudorank of 1; that is, it is singular in machine precision. The result is unchanged when $\mathbf{R}$ is added to $\mathbf{H}\mathbf{P}_0\mathbf{H}^{\mathrm{T}}$. In this case, then, the filter observational update fails because the matrix $\mathbf{H}\mathbf{P}_0\mathbf{H}^{\mathrm{T}} + \mathbf{R}$ is not invertible.

10.9.3.2 Alternative Implementations The covariance correction process (observational update) in the solution of the Riccati equation was found to be the dominant source of numerical instability in the Kalman filter implementation, with the more common symptoms of failure being asymmetry of the covariance matrix (easily fixed) or (worse by far) negative terms on its diagonal. These implementation problems could be avoided for some problems by using more precision, but they were not solved for most applications until James Potter[11] (1937–2005) discovered what is now called " **square root**

[11]In 1962, Potter was a graduate student at MIT, working part-time at the Instrumentation Laboratory on guidance and navigation system development for the Apollo moon project. The Kalman filter for Apollo space navigation would have to be implemented in 15-bit arithmetic, and it was already experiencing numerical stability problems in 36-bit floating-point implementations on a "mainframe" IBM computer. Potter took the problem home with him on a Friday and showed up Monday morning with his solution.

TABLE 10.11. Compatible Methods for Solving the Riccati Equation

	Riccati Equation Implementation Methods	
Covariance Matrix Format	Corrector	Predictor
Symm. nonneg. def. **P**	Kalman [16]	Kalman [16]
	Joseph [6]	Kalman [16]
Square Cholesky factor **C**	Potter [23]	$\mathbf{C}_{k+1-}=\mathbf{\Phi}_k C_{k+}$
Triangular Cholesky factor **C**	Carlson [7]	Kailath–Schmidt[a]
Triangular Cholesky factor **C**	Morf–Kailath combined [22]	
Modified Cholesky factors **U**, **D**	Bierman [3]	Thornton [28]

[a]Unpublished.

filtering." Potter rederived the observational update of **P** using a **Cholesky factor C** of **P** as the dependent variable, where $\mathbf{P} = \mathbf{C}\mathbf{C}^{\mathrm{T}}$. (A true "square root" of **P** would satisfy the alternative equation $\mathbf{P} = \mathbf{C}\mathbf{C}$.)

Soon after Potter's discovery, alternative Cholesky factorizations have been used, as well. Each of these methods requires a compatible method for covariance prediction. Table 10.11 lists several of these compatible implementation methods for improving the numerical stability of Kalman filters.

Figure 10.18 illustrates how these methods perform on the ill-conditioned problem of Example 10.9 as the conditioning parameter $\delta \to 0$. For this particular test case, using 64-bit floating-point precision (52-bit mantissa), the accuracy of the Carlson [7] and Bierman [3] implementations degrade more gracefully than the others as $\delta \to \varepsilon$, the machine precision limit. The Carlson and Bierman solutions still maintain about nine digits (≈30 bits) of accuracy at $\delta \approx \sqrt{\varepsilon}$ when the other methods have essentially no bits of accuracy in the computed solution.

These results, by themselves, do not prove the general superiority of the Carlson and Bierman solutions for the Riccati equation. Relative performance of alternative implementation methods may depend upon details of the specific application, and—for many applications—the standard Kalman filter implementation will suffice. For many other applications, it has been found sufficient to constrain the covariance matrix to remain symmetric.

The MATLAB® m-file `shootout.m` on the accompanying website generates Fig. 10.18, using m-files with the same names as those of the solution methods in Fig. 10.18. For derivations of these methods, see Ref. 12.

10.9.3.3 Conditioning and Scaling Considerations The data formatting differences between triangular Cholesky factors (Carlson implementation) and modified Cholesky factors (Bierman–Thornton implementation) are not always insignificant, as illustrated by the following example.

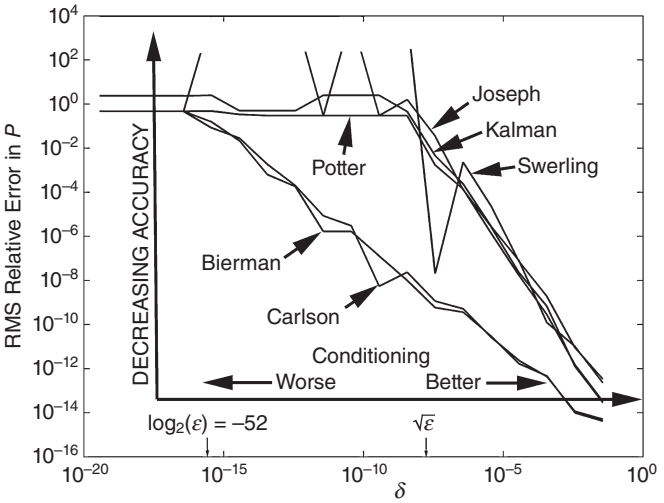


Fig. 10.18 Degradation of numerical solutions with problem conditioning.

Example 10.10 Cholesky Factor Scaling and Conditioning. The $n \times n$ covariance matrix

$$\mathbf{P} = \begin{bmatrix} 10^0 & 0 & 0 & \cdots & 0 \\ 0 & 10^{-2} & 0 & \cdots & 0 \\ 0 & 0 & 10^{-4} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 10^{2-2n} \end{bmatrix} \tag{10.171}$$

has condition number 10^{2n-2}. Its Cholesky factor

$$\mathbf{C} = \begin{bmatrix} 10^0 & 0 & 0 & \cdots & 0 \\ 0 & 10^{-1} & 0 & \cdots & 0 \\ 0 & 0 & 10^{-2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 10^{1-n} \end{bmatrix} \tag{10.172}$$

has condition number 10^{n-1}. However, its modified Cholesky factors are $\mathbf{U} = \mathbf{I}_n$ (condition number = 1) and $\mathbf{D} = \mathbf{P}$ (condition number = 10^{2n-2}).

The condition numbers of the different factors are plotted versus matrix dimension n in Fig. 10.19. As a rule, one would like matrix condition numbers to be $<1/\varepsilon$, where ε is the machine precision limit (the smallest number such that $1 + \varepsilon > 1$ in machine precision, equal to 2^{-52} in the IEEE 64-bit

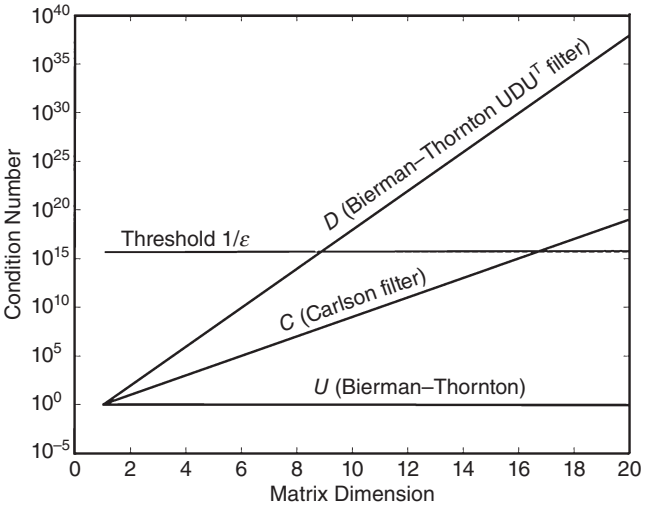


Fig. 10.19 Example 10.10: Conditioning of Cholesky factors.

precision used by MATLAB® on most PCs). This threshold is labeled $1/\varepsilon$ on the plot.

If the implementation is to be done in fixed-point arithmetic, scaling also becomes important. For this example, the nonzero elements of **D** and **C** will have the same relative dynamic ranges as the condition numbers.

10.9.4 Kalman Filter Monitoring

10.9.4.1 Rejecting Anomalous Sensor Data Anomalous sensor data can result from sensor failures or from corruption of the signals from sensors, and it is important to detect these events before the anomalous data corrupts the estimate. The filter is not designed to accept errors due to sensor failures or signal corruption, and they can seriously degrade the accuracy of estimates. The Kalman filter has infinite impulse response, so errors of this sort can persist for some time.

Detecting Anomalous Sensor Data Fortunately, the Kalman filter implementation includes parameters that can be used to detect anomalous data. The Kalman gain matrix

$$\bar{\mathbf{K}}_k = \mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}} \underbrace{(\mathbf{H}_k\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}} + \mathbf{R}_k)^{-1}}_{\mathbf{Y}_{vk}} \tag{10.173}$$

includes the factor

$$\mathbf{Y}_{vk} = (\mathbf{H}_k\mathbf{P}_k(-)\mathbf{H}_k^{\mathrm{T}} + \mathbf{R}_k)^{-1}, \tag{10.174}$$

the information matrix of innovations.[12] The innovations are the measurement residuals

$$\nu_k \overset{\text{def}}{=} \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k(-), \tag{10.175}$$

the differences between the apparent sensor outputs and the predicted sensor outputs. The associated likelihood function for innovations is

$$\mathcal{L}(\nu_k) = \exp\left(-\frac{1}{2}\nu_k^{\mathrm{T}} \mathbf{Y}_{\nu k} \nu_k\right), \tag{10.176}$$

and the log-likelihood is

$$\log[\mathcal{L}(\nu_k)] = -\nu_k^{\mathrm{T}} \mathbf{Y}_{\nu k} \nu_k, \tag{10.177}$$

which can easily be calculated. The equivalent statistic

$$\chi^2 = \frac{\nu_k^{\mathrm{T}} \mathbf{Y}_{\nu k} \nu_k}{\ell} \tag{10.178}$$

(i.e., without the sign change and division by 2, but divided by the dimension of $\mathbf{v}_k$) is nonnegative with a minimum value of zero. If the Kalman filter were perfectly modeled and all white-noise sources were Gaussian, this would be a chi-squared statistic with distribution as plotted in Fig. 10.20. An upper limit threshold value on χ^2 can be used to detect anomalous sensor data, but a practical value of that threshold should be determined by the operational values of χ^2, not the theoretical values; that is, first its range of values should be determined by monitoring the system in operation, then a threshold value $\chi^2_{\max}$ chosen such that the fraction of good data rejected when $\chi^2 > \chi^2_{\max}$ will be acceptable.

Exception Handling for Anomalous Sensor Data The log-likelihood test can be used to detect and reject anomalous data, but it can also be important to use the measurement innovations in other ways:

1. as a minimum, to raise an alarm whenever something anomalous has been detected;
2. to tally the relative frequency of sensor data anomalies, so that trending or incipient failure may be detectable; and
3. to aid in identifying the source, such as which sensor or system may have failed.

[12]Thomas Kailath introduced the notation using the Greek letter ν ("nu") for innovations, because they represent "what is new" in the measurement.

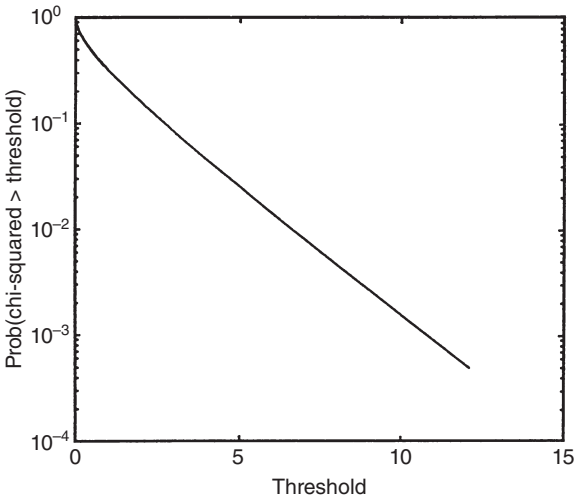


Fig. 10.20 Chi-squared distribution.

10.9.4.2 Monitoring Filter Health Filter health monitoring methods are useful for detecting disparities between the physical system and the model of the system used in Kalman filtering (useful in filter development), for detecting numerical instabilities in the solution of the Riccati equation, and for detecting the onset of poor observability conditions. We have discussed in Section 10.9.4 the monitoring methods for detecting when sensors fail, or for detecting gradual degradation of sensors.

Covariance Analysis Covariance analysis in this context means monitoring selected diagonal terms of the covariance matrix **P** of estimation uncertainty. These are the variances of state estimation uncertainty. System requirements are often specified in terms of the variance or RMS uncertainties of key state variables, and this is a way of checking that these requirements are being met. It is not always possible to cover all operational trajectories in the design of the sensor system. It is possible that situations can occur when these requirements are not being met in operation, and it can be useful to know that.

Checking Covariance Symmetry The so-called square-root filtering methods presented in Section 10.9.3 are designed to ensure that the covariance matrix of estimation uncertainty (the dependent variable of the matrix Riccati equation) remains symmetric and positive definite. Otherwise, the fidelity of the solution of the Riccati equation can degrade to the point that it corrupts the Kalman gain and that it can corrupt the estimate. If you should choose not to use square-root filtering, then you may need some assurance that the decision was justified.

Verhaegen and Van Dooren [29] have shown that asymmetry of $\mathbf{P}$ is one of the factors contributing to numerical instability of the Riccati equation. If square-root filtering is not used, then the covariance matrix can be "symmetrized" occasionally by adding it to its transpose and rescaling:

$$\mathbf{P} := \frac{1}{2}(\mathbf{P} + \mathbf{P}^{\mathrm{T}}). \tag{10.179}$$

This trick has been used for many years to head off numerical instabilities.

Checking the Means and Autocorrelations of Innovations Innovations are the differences between what comes out of the sensors and what was expected, based on the estimated system state. If the system were perfectly modeled in the Kalman filter, the innovations would be a zero-mean white-noise process and its autocorrelation function would be zero except at zero delay. The departure of the empirical autocorrelation of innovations from this model is a useful tool for analysis of mismodeling in real-world applications.

CALCULATION OF AUTOCOVARIANCE AND AUTOCORRELATION FUNCTIONS The mean of the innovations should be zero. If not, the mean must be subtracted from the innovations before calculating the autocovariance and autocorrelation functions of the innovations.

For vector-valued innovations, the autocovariance function is a matrix-valued function, defined as

$$\mathbf{A}_{\mathrm{covar},k} \overset{\mathrm{def}}{=} E_i\left\langle \Delta\mathbf{z}_{v,i} \Delta\mathbf{z}_{v,i+k}^{\mathrm{T}} \right\rangle, \tag{10.180}$$

$$\Delta\mathbf{z}_{v,i} \overset{\mathrm{def}}{=} \mathbf{z}_i - \mathbf{H}_i\mathbf{x}_i(-)\,(\text{innovations}), \tag{10.181}$$

and the autocorrelation function is defined by

$$\mathbf{A}_{\mathrm{correl},k} \overset{\mathrm{def}}{=} \mathbf{D}_\sigma^{-1}\mathbf{A}_{\mathrm{covar},k}\mathbf{D}_\sigma^{-1}, \tag{10.182}$$

$$\mathbf{D}_\sigma \overset{\mathrm{def}}{=} \begin{bmatrix} \sigma_1 & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2 & 0 & \cdots & 0 \\ 0 & 0 & \sigma_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \sigma_\ell \end{bmatrix}, \tag{10.183}$$

$$\sigma_j \overset{\mathrm{def}}{=} \sqrt{\{\mathbf{A}_{\mathrm{covar},0}\}_{jj}}, \tag{10.184}$$

where the jth diagonal element $\{\mathbf{A}_{\text{covar},0}\}_{jj}$ of $\mathbf{A}_{\text{covar},0}$ is the variance of the jth component of the innovations vector.

Calculation of Spectra and Cross Spectra The Fourier transforms of the diagonal elements of the autocovariance function $\mathbf{A}_{\text{covar},k}$ (i.e., as functions of k) are the power spectral densities (spectra) of the corresponding components of the innovations, and the Fourier transforms of the off-diagonal elements are the cross spectra between the respective components.

INTERPRETATION OF RESULTS Simple patterns to look for include the following:

1. Nonzero means of innovations may indicate the presence of uncompensated sensor output biases or mismodeled output biases. The modeled variance of the bias may be seriously underestimated, for example.
2. Innovations means increasing or varying with time may indicate output noise that is a random walk or an exponentially correlated process.
3. Exponential decay of the autocorrelation functions is a reasonable indication of unmodeled (or mismodeled) random walk or exponentially correlated noise.
4. Spectral peaks may indicate unmodeled harmonic noise, but it could also indicate that there is an unmodeled harmonic term in the state dynamic model.
5. The autocovariance function at zero delay, $\mathbf{A}_{\text{covar},0}$, should equal $\mathbf{HPH}^{\mathrm{T}} + \mathbf{R}$ for time-invariant or very slowly time-varying systems. If $\mathbf{A}_{\text{covar},0}$ is much bigger than $\mathbf{HPH}^{\mathrm{T}} + \mathbf{R}$, it could indicate that $\mathbf{R}$ is too small or that the process noise $\mathbf{Q}$ is too small, either of which may cause $\mathbf{P}$ to be too small. If $\mathbf{A}_{\text{covar},0}$ is much smaller than $\mathbf{HPH}^{\mathrm{T}} + \mathbf{R}$, $\mathbf{R}$ and/or $\mathbf{Q}$ may be too large.
6. If the off-diagonal elements of $\mathbf{A}_{\text{correl},0}$ are much bigger than those of $\mathbf{D}_{\sigma}^{-1}\left(\mathbf{HPH}^{\mathrm{T}} + \mathbf{R}\right)\mathbf{D}_{\sigma}^{-1}$, then there may be unmodeled correlations between sensor outputs. These correlations could be caused by mechanical vibration or power supply noise, for example.

10.10 SUMMARY

Although there is a vast literature on Kalman filtering theory and implementation methods, the material presented in this chapter should cover most of the important issues for applying Kalman filtering to GNSS navigation and GNSS/INS integration. References for additional coverage have been cited, where appropriate. Although there are new methods for implementing Kalman filtering to highly nonlinear applications, the GNSS and INS navigation problems are generally near enough to being linear that such methods are not required.

Illustrative applications have been demonstrated, and the necessary algorithms have been presented and implemented in MATLAB® for you to

experience on your own. These include demonstrations of GNSS navigation, alternative models for host vehicle dynamics, methods for assessing how performance depends on the choice of vehicle models, and results to demonstrate the importance of model fidelity.

In order to apply Kalman filtering effectively, it is important to understand how the Kalman filter works—and to recognize when it is not working properly. For that reason, we have included methods for monitoring and analyzing Kalman filter performance during operation. Common engineering practice should require all Kalman filter applications to be thoroughly understood and verified before they can be released.

PROBLEMS

10.1 Given the scalar plant and observation equations

$$x_k = x_{k-1}, z_k = x_k + v_k \sim N(0, \sigma_{v^2})$$

and white noise

$$\mathrm{E}x_0 = 1, \mathrm{E}x_0^2 = \mathbf{P}_0,$$

find the estimate of x_k and the steady-state covariance.

10.2 Given the vector plant and scalar observation equations,

$$x_k = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_{k-1} + w_{k-1} \text{ (normal and white)},$$

$$z_k = [1\ 0]x_k + v_k, \text{(normal and white)},$$

$$Ew_k = 0, Q_k = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.$$

$$Ev_k = 0, R_k = 1 + (-1)^k,$$

find the covariances and Kalman gains for $k = 10, \mathbf{P}_0 = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$

10.3 Given

$$x_k = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_{k-1} + \begin{bmatrix} 1 \\ \frac{1}{2} \end{bmatrix} (-g),$$

$$z_k = [1\ 0]x_k + v_k \sim \text{normal and white},$$

where g is gravity, find $\hat{x}_k$, $\mathbf{P}_k(+)$ for $k = 6$:

$$\hat{x}_0 = \begin{bmatrix} 90 \\ 1 \end{bmatrix}, \mathbf{P}_0 = \begin{bmatrix} 10 & 0 \\ 0 & 2 \end{bmatrix},$$

$$Ev_k = 0, Ev_k^2 = 2.$$

10.4 Given

$$x_k = -2x_{k-1} + w_{k-1},$$

$$z_k = x_k + v_k \sim \text{normal and white},$$

$$Ev_k = 0, Ev_k^2 = 1,$$

$$Ew_k = 0, E(w_k w_j) = e^{-|k-j|},$$

find the covariances and Kalman gains for $k = 3$, $\mathbf{P}_0 = 10$.

10.5 Given $E[(w_k - 1)(w_j - 1)] = e^{-|k-j|}$ find the discrete equation model.

10.6 Given $E\langle[w(t_1) - 1][w(t_2) - 1]\rangle = e^{-|t_1 - t_2|}$ find the differential equation model.

10.7 Based on the 24-satellite GNSS constellation, five satellite trajectories are selected, and their parameters tabulated accordingly:

α=55°		
Satellite ID	Ω_0 (deg)	Θ_0 (deg)
6	272.847	268.126
7	332.847	80.956
8	32.847	111.876
9	92.847	135.226
10	152.847	197.046

(a) Choose correctly phased satellites of four.

(b) Calculate DOPs to show their selection by plots.

(c) Use Kalman filter equations for $\mathbf{P}_k(-)$, $\bar{K}_k$ and $\mathbf{P}_k(+)$ to show the errors. Draw the plots. This should be done with good geometric dilution of precision (GDOP).

Choose user positions at (0, 0, 0) for simplicity.

Hint: Use `GPS_perf.m` from the accompanying website, Chapter 7. See also Appendix A, Software for Chapter 7.

REFERENCES

[1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. MIT Press, Cambridge, MA, 2006.

[2] D. S. Bernstein, *Matrix Mathematics*. Princeton University Press, Princeton, NJ, 2005.

[3] G. J. Bierman, *Factorization Methods for Discrete Sequential Estimation*, Vol. 128, Mathematics in Science and Engineering. Academic Press, New York, 1977.

[4] D. J. Biezad, *Integrated Navigation and Guidance Systems*. American Institute of Aeronautics and Astronautics, New York, 1999.

[5] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering with Matlab Exercises*, 4th ed. John Wiley & Sons, Hoboken, NJ, 2012.

[6] R. S. Bucy and P. D. Joseph, *Filtering for Stochastic Processes with Applications to Guidance*. Chelsea, New York, 1968 (republished by the American Mathematical Society).

[7] N. A. Carlson, "Fast Triangular Formulation of the Square Root Filter," *AIAA Journal* **11**(9), 1259–1265 (1973).

[8] R. M. du Plessis, "Poor Man's Explanation of Kalman Filtering, or How I Stopped Worrying and Learned to Love Matrix Inversion," Autonetics Technical Note, Anaheim, CA, 1967, republished by Taygeta Scientific Incorporated, Monterey, CA, 1996.

[9] P. Dyer and S. McReynolds, "Extension of Square-Root Filtering to Include Process Noise," *Journal of Optimization Theory and Applications* **3**, 444–458 (1969).

[10] A. Gelb (Ed.), *Applied Optimal Estimation*. MIT Press, Cambridge, MA, 1974.

[11] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Johns Hopkins University Press, Baltimore, MD, 1996.

[12] M. S. Grewal and A. P. Andrews, *Kalman Filtering Theory and Practice Using MATLAB*, 3rd ed. John Wiley & Sons, Hoboken, NJ, 2008.

[13] H. V. Henderson and S. R. Searle, "On Deriving the Inverse of a Sum of Matrices," *SIAM Review* **23**, 53–60 (1981).

[14] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. Dover Publications, New York, 2009 (reprint of 1970 edition by Academic Press).

[15] T. Kailath, A. H. Sayed, and B. Hassibi, "Kalman Filtering Techniques," in J. G. Webster (Ed.), *Wiley Encyclopedia of Electrical and Electronics Engineering*. John Wiley & Sons, Hoboken, NJ, 1999.

[16] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME, Series D: Journal of Basic Engineering* **82**, 35–45 (1960).

[17] R. E. Kalman and R. S. Bucy, "New Results in Linear Filtering and Prediction Theory," *Transactions of the ASME, Series D: Journal of Basic Engineering* **83**, 95–108 (1961).

[18] P. G. Kaminski, *Square Root Filtering and Smoothing for Discrete Processes*, PhD Thesis, Stanford University, Stanford, CA, 1971.

[19] P. Langevin, "Sur la théorie du muovement brownien," *Comptes Rendus de l'Académie des Sciences* Paris **146**, 530–533 (1908).

[20] L. J. Levy, "The Kalman Filter: Integrations's Workhorse," *GPS World*, September 1977, pp. 65–71.

[21] L. A. McGee and S. F. Schmidt, *Discovery of the Kalman Filter as a Practical Tool for Aerospace and Industry*, National Aeronautics and Space Administration report NASA-TM-86847, 1985.

[22] M. Morf and T. Kailath, "Square Root Algorithms for Least Squares Estimation," *IEEE Transactions on Automatic Control* **AC-20**, 487–497 (1975).

[23] J. E. Potter and R. G. Stern, "Statistical Filtering of Space Navigation Measurements," *Proceedings of the 1963 AIAA Guidance and Control Conference*, American Institute of Aeronautics & Astronautics, Washington, DC, 1963.

[24] K. C. Redmond and T. M. Smith, *From Whirlwind to MITRE: The R&D Story of the SAGE Air Defense Computer*. MIT Press, Cambridge, MA, 2000.

[25] J. F. Riccati, "Animadversationnes in aequationes differentiales secundi gradus," *Acta Eruditorum Quae Lipside Publicantur Supplimenta* **8**, 66–73 (1724).

[26] S. F. Schmidt, "Application of State Space Methods to Navigation Problems," in C. T. Leondes (Ed.), *Advances in Control Systems*, Vol. 3. Academic Press, New York, 1966.

[27] D. Simon, *Optimal State Estimation*. John Wiley & Sons, Hoboken, NJ, 2006.

[28] C. L. Thornton, *Triangular Covariance Factorizations for Kalman Filtering*, PhD Thesis, University of California at Los Angeles, School of Engineering, 1976.

[29] M. Verhaegen and P. Van Dooren, "Numerical Aspects of Different Kalman Filter Implementations," *IEEE Transactions on Automatic Control* **AC-31**, 907–917 (1986).

[30] C. E. Wheatley, III, C. G. Mosley, and E. V. Hunt, *Controlled Oscillator Having Random Variable Frequency*, U.S. Patent No. 4,646,032, February 24, 1987.

[31] P. Zarchan and H. Musoff, *Fundamentals of Kalman Filtering: A Practical Approach*, Vol. 190. Progress in Aeronautics and Astronautics. AIAA, New York, 2000.

11

INERTIAL NAVIGATION ERROR ANALYSIS

Errors, like straws, upon the surface flow;
He who would search for pearls, must dive below.
—John Dryden, "All for Love," 1678

11.1 CHAPTER FOCUS

The purpose of this chapter is to show where mathematical models characterizing the dynamics of inertial navigation system (INS) navigation errors come from and how these models can be used for characterizing INS performance. In the next chapter, these same models will be augmented with global navigation satellite system (GNSS) error models for implementing GNSS/INS integration.

Many of these models were developed in the 1960s for analyzing how inertial sensor errors influence navigation performance and for integrating inertial systems with auxiliary sensors to improve overall performance. The general subject has been called "systems analysis," and it has become a powerful tool for the development of very complex sensing and estimation systems, including GNSS.

Systems analysis of this sort allows designers to predict the performance of large, complex systems using various sensors with various individual

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

performance statistics. It played a major role in the design and implementation of the navigation system for the Apollo missions to the moon and back at the end of the 1960s. One of the textbooks on inertial navigation systems analysis from that era was recently republished [1].[1] An early pioneer in this effort was Stanley F. Schmidt [4], who had also pioneered in the practical development and application of Kalman filtering.

The focus here is on the same sorts of models and methods used in Chapter 10 for assessing the expected GNSS receiver performance on representative mission trajectories, but specialized for INS errors. These linear stochastic system models represent uncertainty propagation for inertial navigation systems, with and without using auxiliary sensors.

How this type of modeling should be done depends on (1) the error characteristics of the intended sensors and (2) the dynamic conditions and navigation accuracy requirements for the intended INS missions. The models developed here are intended to demonstrate how modeling is done, but the result is not a one-size-fits-all model for all GNSS/INS integration. It is intended to demonstrate a technical approach to GNSS/INS integration.

Order of presentation:

1. Define variables representing errors in the navigation solution. For inertial navigation, these must represent errors in location, velocity, and orientation.
2. Show how these errors become dynamically coupled through the calculations of the navigation solution in the terrestrial environment.
3. Show how this coupling can be modeled as a linear time-varying dynamic system representing the propagation of navigation errors over time and how this model depends on the INS trajectory.
4. Using that model, show why inertial navigation errors are unstable in the vertical direction and how this can be mitigated by using an altimeter as an auxiliary sensor for applications that require vertical navigation. However, vertical navigation can be eliminated for surface ships, which reduces the number of essential navigation error variables from 9 to 7.
5. Show how Schuler oscillations and the Coriolis effect shape navigation error trajectories.
6. Show how the effects of sensor noise can be modeled as a time-varying linear stochastic system representing stand-alone performance of the INS, and show how this model relates noise levels to INS performance.
7. Show how sensor compensation errors also contribute to navigation errors, how the basic navigation error model can be augmented to include these effects, and how the resulting model characterizes the effects of drifting sensor compensation parameters.

[1]There is another classic treatise on the subject by Widnall and Grundy [8], but it is long out of print.

11.2 ERRORS IN THE NAVIGATION SOLUTION

The "navigation solution" for inertial navigation is what it takes to propagate the initial navigation solution forward in time, starting with the initial conditions of location, velocity, and orientation. These initial conditions are never known to infinite precision, however, and the forward propagation of the solution is implemented in finite precision. As a result, there will always be small errors in the navigation solution. It turns out that these errors also have distinctive ways of corrupting the calculations in the solution implementation, so that these errors have their own dynamic evolution model.

This section is about that model.

11.2.1 The Nine Core INS Error Variables

For inertial navigation, navigation errors include errors in location, velocity, and orientation. As described in Section 3.7.2 of Chapter 3, these are the same variables required for initializing the navigation solution, and they are used to carry the navigation solution forward in time. There are three components of each, so the minimum dimension of the navigation error state vector will be nine. These nine "core" inertial navigation error variables will also define how navigation errors propagate over time.

11.2.2 Coordinates Used for INS Error Analysis

INS analysis can be complicated in any coordinate system. It was originally developed using locally level coordinates because early systems were gimbaled with locally level inertial measurement unit (IMU) axes for navigation. This approach was natural in some ways because INS performance requirements were specified in locally level coordinates. It is also natural because locally level coordinates represent the direction of gravity, which is very important for inertial navigation in the terrestrial environment. Locally level coordinates are used here for the same reasons. These could be either north–east–down (NED) or east–north–up (ENU). ENU coordinates are used here, although NED coordinates have been used elsewhere. Derivations in NED coordinates would follow similar lines. Derivations with respect to latitude and longitude are not that dissimilar.

11.2.3 Model Variables and Parameters

Table 11.1 is a list of symbols and definitions of the parameters and variables used in the modeling and derivations of navigation error dynamics. These include nine navigation error variables in ENU coordinates, and other variables and parameters used in the INS implementation. The "hatted" variables (e.g., $\hat{\phi}$) represent values used in the INS implementation.

TABLE 11.1. List of Symbols and Approximations

Earth Model Parameters[a]					
$\bar{R}_\oplus$	$\stackrel{\text{def}}{=}$	Mean radius			$\approx 0.6371009 \times 10^7$ (m)
$a_\oplus$	$\stackrel{\text{def}}{=}$	Equatorial radius			$\approx 0.6378137 \times 10^7$ (m)
$f_\oplus$	$\stackrel{\text{def}}{=}$	Flattening			$\approx 1/298.25722356$ ≈ 0.00335281066475
$\mathrm{GM}_\oplus$	$\stackrel{\text{def}}{=}$	Gravity constant			$\approx 0.3986004 \times 10^{15}$ ($\mathrm{m}^3/\mathrm{s}^2$)
$\Omega_\oplus$	$\stackrel{\text{def}}{=}$	Rotation rate			$\approx 0.7292115 \times 10^{-4}$ (rad/s)
INS Navigation Solution					
$\hat{\phi}$	=	ϕ	+	$\varepsilon_\mathrm{N} / \bar{R}_\oplus$	Latitude (rad)
$\hat{\theta}$	=	θ	+	$\varepsilon_\mathrm{E} / (\bar{R}_\oplus \cos\phi)$	Longitude (rad)
$\hat{E}$	=	E	+	ε_E	Easting with respect to INS (m)
$\hat{N}$	=	N	+	ε_N	Northing with respect to INS (m)
$\hat{h}$	=	h	+	ε_U	Altitude (m)
$\hat{v}_\mathrm{E}$	=	v_E	+	$\dot{\varepsilon}_\mathrm{E}$	East INS velocity (m/s)
$\hat{v}_\mathrm{N}$	=	v_N	+	$\dot{\varepsilon}_\mathrm{N}$	North INS velocity (m/s)
$\hat{v}_\mathrm{U}$	=	v_U	+	$\dot{\varepsilon}_\mathrm{U}$	Vertical INS velocity (m/s)
$\mathbf{C}_{\mathrm{ENU}}^{\mathrm{INS}}$	$\approx$	$\mathbf{I}$	+	$\boldsymbol{\rho}\otimes$	Coordinate transformation matrix, INS to ENU
ρ_E	$\stackrel{\text{def}}{=}$	INS misalignment about east axis (rad)			
ρ_N	$\stackrel{\text{def}}{=}$	INS misalignment about north axis (rad)			
ρ_U	$\stackrel{\text{def}}{=}$	INS misalignment about vertical axis (rad)			
Miscellaneous Variables and Symbols					
ψ	=	Horizontal velocity direction, measured counterclockwise from east (rad)			
$\boldsymbol{\omega}_\oplus$	$\stackrel{\text{def}}{=}$	Earthrate vector (rad/s)			
$\boldsymbol{x}$	=	Vector from the center of the earth to the INS			
$\otimes$	$\stackrel{\text{def}}{=}$	Vector cross product; as a suffix, it transforms a vector into its equivalent skew-symmetric matrix			

[a] $\oplus$ is the astronomical symbol for Earth.

All error variables are represented in ENU coordinates centered at the actual INS location.

The state variable representations of the nine core INS errors are listed in Table 11.2.

11.2.3.1 INS Orientation Variables and Errors INS orientation with respect to gravity and the earth rotation axis is typically represented in terms of the coordinate transformation between the INS sensor axes and locally level

TABLE 11.2. State Variables for the Nine Core INS Errors

State Vector			
$\boldsymbol{\xi}$	$\stackrel{\text{def}}{=}$	$\begin{bmatrix} \boldsymbol{\varepsilon} \\ \dot{\boldsymbol{\varepsilon}} \\ \boldsymbol{\rho} \end{bmatrix}$	INS navigation error
Subvectors			
$\boldsymbol{\varepsilon}$	$\stackrel{\text{def}}{=}$	$\begin{bmatrix} \varepsilon_{\mathrm{E}} \\ \varepsilon_{\mathrm{N}} \\ \varepsilon_{\mathrm{U}} \end{bmatrix}$	INS location error
$\dot{\boldsymbol{\varepsilon}}$	$\stackrel{\text{def}}{=}$	$\begin{bmatrix} \dot{\varepsilon}_{\mathrm{E}} \\ \dot{\varepsilon}_{\mathrm{N}} \\ \dot{\varepsilon}_{\mathrm{U}} \end{bmatrix}$	INS velocity error
$\boldsymbol{\rho}$	$\stackrel{\text{def}}{=}$	$\begin{bmatrix} \rho_{\mathrm{E}} \\ \rho_{\mathrm{N}} \\ \rho_{\mathrm{U}} \end{bmatrix}$	INS orientation error

ENU coordinates. It will simplify the derivation somewhat if we assume that the INS sensor axes are nominally parallel to the ENU axes, except for small orientation errors. By "small," we mean that they can be represented by rotations in the order of a milliradian or less. At those levels, error dynamics can be modeled in terms of first-order variations, and second-order effects can be ignored.

Misalignments and Tilts Misalignment variables are used to represent INS orientation errors. These are different enough from the other state variables that some clarification about what they mean and how they are used may be useful.

Misalignments represent the rotational difference between these locally level reference directions and what the navigation solution has estimated for them.

Error in INS-calculated directions can be divided into categories:

1. INS misalignments with respect to locally level coordinates at the actual location of the INS. These cause errors in the calculation of
   (a) Gravity, which is needed for navigation in an accelerating coordinate frame. Orientation errors involved in the miscalculation errors are sometimes called "tilt errors" because they are equivalent to rotating the gravity vector about horizontal axes.

(b) The direction of the rotation axis of the earth, which is needed for navigation in a rotating coordinate frame.

2. Errors in the INS estimate of its location. Errors in the estimated latitude, in particular, cause errors in the expected elevation of the earth's rotation axis above or below the horizon. These are not true misalignments, but their effects must be taken into account when determining the dynamic cross coupling between different error types. Errors in longitude do not have much influence on dynamic coupling of navigation errors, although they do influence pointing accuracies with respect to objects in space.

Effect of INS Misalignments INS misalignments are represented in terms of a coordinate transformation between what the INS *believes* to be locally level ENU coordinates, and the actual locally level ENU coordinates. If the misalignments are "small" (in the order of milliradian or less) the coordinate transformation $\mathbf{C}_{\text{ENU}}^{\text{INS}}$ from INS coordinates to ENU coordinates can be approximated as

$$\mathbf{C}_{\text{ENU}}^{\text{INS}}(\boldsymbol{\rho}) \approx \mathbf{I} + \boldsymbol{\rho}\otimes \tag{11.1}$$

$$= \begin{bmatrix} 1 & -\rho_{\text{U}} & \rho_{\text{N}} \\ \rho_{\text{U}} & 1 & -\rho_{\text{E}} \\ -\rho_{\text{N}} & \rho_{\text{E}} & 1 \end{bmatrix}. \tag{11.2}$$

This approximation is the first-order term in the series expansion of the matrix exponential of the skew-symmetric matrix $\boldsymbol{\rho}\otimes$, which is the exact form of a coordinate transformation equivalent to a rotation. This small-angle approximation is used where necessary to transform INS variables to ENU coordinates.

Figure 11.1 shows a misalignment vector $\boldsymbol{\rho}$ representing a small-angle[2] rotation of the reference navigation coordinates used by the INS. True ENU coordinate axes at the location of the INS are labeled *E*, *N*, and *U*. However, the orientation error of the INS is such that its estimated coordinate directions are actually those labeled $\hat{E}$, $\hat{N}$, and $\hat{U}$ in the figure. This effective coordinate change is equivalent to rotation about the vector $\boldsymbol{\rho}$ by a small angle $|\boldsymbol{\rho}|$.

Misalignments contribute to navigation errors in a number of ways. Figure 11.1 illustrates how misalignments cause miscalculation of the value of earth rotation used in the INS for maintaining its reference axes locally level, and how northing errors compound the error. The actual rotation rate vector in true ENU coordinates is labeled $\boldsymbol{\omega}_{\oplus}$ in the figure. However, due to

[2]For illustrative purposes only, the actual rotation angle in the figure has been made relatively large.

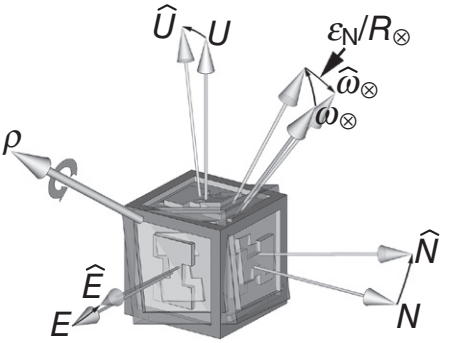


Fig. 11.1 Misalignment of INS navigation coordinates.

misalignment error, the INS representation of this direction has been rotated twice: once by misalignments and again by miscalculation of latitude due to northing error ε_N divided by $\bar{R}_\oplus$, the mean radius of the earth. As a consequence, in true ENU coordinates, the actual earth rotation rate vector (labeled $\boldsymbol{\omega}_\oplus$ in the figure) is moved to the direction labeled $\hat{\boldsymbol{\omega}}_\oplus$ in the figure by INS navigation errors.

The resulting estimated value $\hat{\boldsymbol{\omega}}_\oplus$ is used in the INS for maintaining its reference navigation coordinate aligned with what it believes to be true ENU coordinates and for compensating for the Coriolis effect. The Coriolis implementation will have further compounding effects due to errors in estimated velocity. The result is a fairly complex structure of how navigation errors cause even more errors through how inertial navigation is implemented. This complexity is reduced somewhat by using only first-order error modeling.

In the first-order analysis model, what the small-angle approximation does to any vector ν is approximated by

$$\hat{\boldsymbol{\nu}} \approx [\mathbf{I} + \boldsymbol{\rho}\otimes]\boldsymbol{\nu} \tag{11.3}$$

$$= \boldsymbol{\nu} + \boldsymbol{\rho}\otimes\boldsymbol{\nu} \tag{11.4}$$

$$= \boldsymbol{\nu} - \boldsymbol{\nu}\otimes\boldsymbol{\rho} \tag{11.5}$$

$$= \boldsymbol{\nu} + \begin{bmatrix} 0 & \nu_3 & -\nu_2 \\ -\nu_3 & 0 & \nu_1 \\ \nu_2 & -\nu_1 & 0 \end{bmatrix}\boldsymbol{\rho}. \tag{11.6}$$

These formulas are used repeatedly to represent how misalignments corrupt the estimated variables used in the INS implementation and how this affects navigation errors.

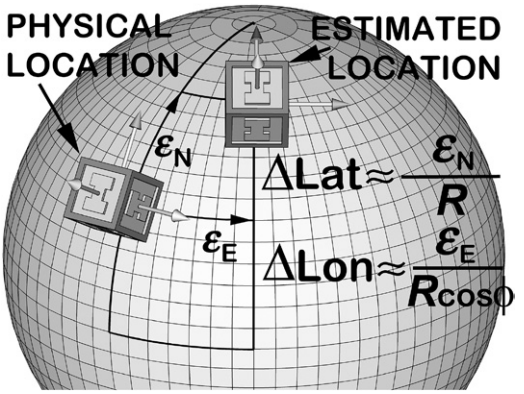


Fig. 11.2 Effects of INS location errors.

Small-Angle Rotation Rate Approximation The differential equation

$$\frac{d}{dt}\mathbf{C} = \mathbf{C}[\boldsymbol{\omega}\otimes] \tag{11.7}$$

models the time rate of change in a coordinate transformation matrix **C** due to a rotation rate vector $\boldsymbol{\omega}$. The formula is generally not recommended for implementing strapdown rotation rate integration, but it is adequate for deriving a first-order model for INS orientation error propagation. Also,

$$\text{if} \quad \mathbf{C} \approx \mathbf{I} + \boldsymbol{\rho}\otimes, \text{ then } \frac{d}{dt}\mathbf{C} \approx \left[\frac{d}{dt}\boldsymbol{\rho}\right]\otimes.$$

Effects of Location Errors Figure 11.2 is an illustration of the effective small-angle rotations associated with horizontal location errors. The north component of location error, ε_N, is equivalent to a small-angle rotation of

$$\Delta_{\text{Lat}} = \frac{\varepsilon_N}{\bar{R}_\oplus} \tag{11.8}$$

about the negative east axis. Consequently, the equivalent small-angle rotation vector would be

$$\tilde{\boldsymbol{\rho}}(\varepsilon_N) \approx \begin{bmatrix} -\dfrac{\varepsilon_N}{\bar{R}_\oplus} \\ 0 \\ 0 \end{bmatrix}. \tag{11.9}$$

The primary effect of north location error is in the miscalculation of latitude, which has a direct effect on the calculation of earthrate. Fortunately, gyrocompass alignment of an INS ordinarily estimates latitude.

The east component of location error, ε_E, is equivalent to a rotation of

$$\Delta_{Lon} = \frac{\varepsilon_E}{\bar{R}_\oplus \cos\phi} \tag{11.10}$$

about the earth's rotation axis (the polar axis), which has no effect on the estimated direction of the earth's rotation axis in locally level ENU coordinates. This change in orientation is equivalent to the small-angle rotation vector

$$\tilde{\boldsymbol{\rho}}(\varepsilon_E) \approx \begin{bmatrix} 0 \\ \dfrac{\varepsilon_E}{\bar{R}_\oplus} \\ \dfrac{\sin\phi\varepsilon_E}{\bar{R}_\oplus \cos\phi} \end{bmatrix}, \tag{11.11}$$

where ϕ is the latitude of the INS location. Initial gyrocompass alignment of an INS makes no determination of longitude. It must be initialized by other means.[3]

The net coordinate rotation effect from location errors can then be represented as

$$\tilde{\boldsymbol{\rho}}(\varepsilon) \approx \begin{bmatrix} -\dfrac{\varepsilon_N}{\bar{R}_\oplus} \\ \dfrac{\varepsilon_E}{\bar{R}_\oplus} \\ \dfrac{\sin\phi\varepsilon_E}{\bar{R}_\oplus \cos\phi} \end{bmatrix}, \tag{11.12}$$

although the last two components have little influence on the propagation of navigation errors.

[3]Erroneous initialization of INS longitude is one explanation offered for events leading to the 1983 destruction of Korean Airlines Flight 007 by Soviet military aircraft after it had passed through Soviet airspace, because an initial error in longitude would have changed its planned route to its destination.

11.2.4 Dynamic Coupling Mechanisms

11.2.4.1 Dynamic Coupling Dynamic coupling of variables is the coupling of one variable into the derivative of another. This happens in inertial navigation because things that cannot be sensed have to be calculated using the estimated values of the navigation solution. This includes gravity, for example, which cannot be sensed by the accelerometers. It must then be estimated using the estimated values of INS location and orientation, and added to the sensed accelerations to calculate INS acceleration relative to the earth. It is through errors in calculating gravity that the other navigation errors become dynamically coupled to velocity errors.

For inertial navigation in locally level coordinates, there are other variables that cannot be sensed directly and must be calculated in the same manner. Figure 11.3 illustrates how the INS calculates and uses such variables. In this flowchart of the INS navigation implementation, the numbered boxes represent six such calculations which use the estimated values of the navigation solution (enclosed in the dashed box) to estimate variables that cannot be

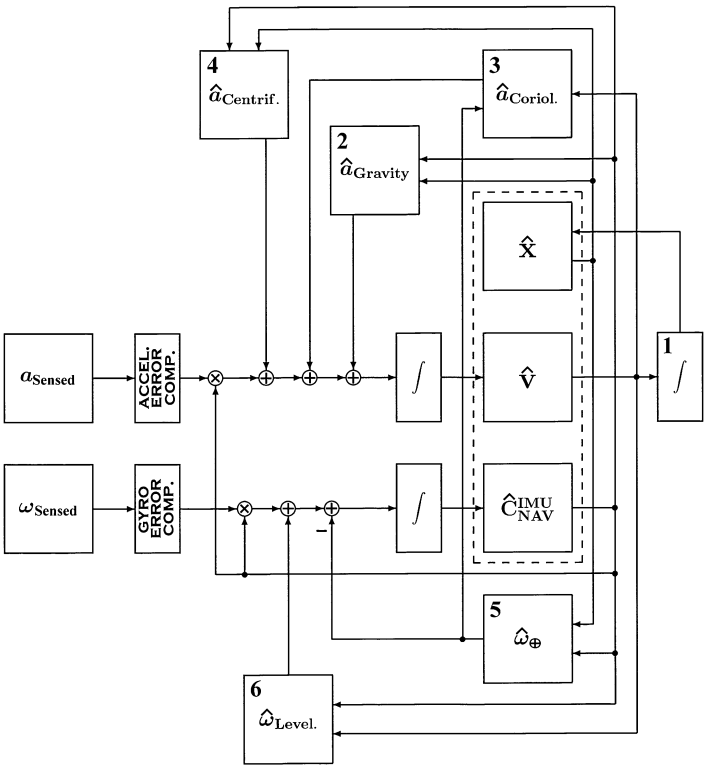


Fig. 11.3 INS navigation solution flowchart.

sensed directly but must be estimated from the navigation solution. These processes include

1. *Integration of Velocities.* INS location is part of the navigation solution, although it cannot be measured directly. It must be inferred from the integrals of the velocities. However, any errors in estimated velocities are also integrated, adding to location errors. This will dynamically couple velocity errors into location errors.
2. *Estimating Gravity.* Gravity cannot be sensed and it is far too big to be ignored. It must be calculated using the navigation solution and integrated along with the sensed accelerations to calculated velocity relative to the earth. Miscalculations due to navigation errors will dynamically couple them into velocity errors.
3. *Coriolis Acceleration.* Coriolis accelerations in rotating coordinates cannot be sensed either. They must be estimated from the navigation solution from the cross product of the velocity vector and the coordinate rotation rate vector. Miscalculation of the Coriolis effect dynamically couples navigation errors into velocity errors.
4. *Centrifugal Acceleration.* This is another type of acceleration in rotating coordinates that cannot be sensed, so it must be estimated from the navigation solution using the estimated values of the earth rotation rate vector and the INS velocity vector in INS coordinates. Miscalculation of centrifugal acceleration due to navigation errors will dynamically couple navigation errors into velocity errors.
5. *Earthrate Compensation in Leveling.* Leveling is the process of maintaining a locally level reference frame for terrestrial navigation. Due to the rotation of the earth, the INS senses rotation rates even when it is stationary with respect to Earth's surface. For navigating in earth-fixed coordinates, the INS must estimate the contribution due to the earth rotation rate based on its navigation solution for latitude and orientation, and subtract it to maintain its locally level orientation. Errors in the navigation solution will corrupt this calculation, which will then dynamically couple location and orientation errors into orientation errors.
6. *Velocity Leveling.* Leveling is also required for maintaining locally level reference directions while the INS moves over the curved surface of the earth. The necessary coordinate rotation rates are calculated using the estimated values of velocity with respect to the earth and estimated orientation. Miscalculation of this correction due to navigation errors will dynamically couple velocity and orientation errors into orientation errors.

In the above listing, note that

1. Only the first of these (velocity integration) dynamically couples other navigation errors into location errors.

2. The next three (gravity, Coriolis, and centrifugal accelerations) dynamically couple navigation errors to velocity errors.
3. The last two (earthrate compensation and leveling) dynamically couple navigation errors to orientation errors.

This partitioning will determine the data structure of the dynamic coefficient matrix in the model for propagating navigation errors. The resulting time-varying linear dynamic model for navigation errors will have the partitioned form

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{\varepsilon}\\ \dot{\boldsymbol{\varepsilon}}\\ \boldsymbol{\rho}\end{bmatrix}=\begin{bmatrix}\mathbf{F}_{11} & \mathbf{F}_{12} & \mathbf{F}_{13}\\ \mathbf{F}_{21} & \mathbf{F}_{22} & \mathbf{F}_{23}\\ \mathbf{F}_{31} & \mathbf{F}_{32} & \mathbf{F}_{33}\end{bmatrix}\begin{bmatrix}\boldsymbol{\varepsilon}\\ \dot{\boldsymbol{\varepsilon}}\\ \boldsymbol{\rho}\end{bmatrix}, \tag{11.13}$$

where

$\mathbf{F}_{11}$ represents the dynamic coupling of location errors into location errors,
$\mathbf{F}_{12}$ represents the dynamic coupling of velocity errors into location errors,
$\mathbf{F}_{13}$ represents the dynamic coupling of orientation errors into location errors,
$\mathbf{F}_{21}$ represents the dynamic coupling of location errors into velocity errors,
$\mathbf{F}_{22}$ represents the dynamic coupling of velocity errors into velocity errors,
$\mathbf{F}_{23}$ represents the dynamic coupling of orientation errors into velocity errors,
$\mathbf{F}_{31}$ represents the dynamic coupling of location errors into orientation errors,
$\mathbf{F}_{32}$ represents the dynamic coupling of velocity errors into orientation errors,
$\mathbf{F}_{33}$ represents the dynamic coupling of orientation errors into orientation errors,

In the next section, mathematical models for each of the six navigation calculations labeled in Fig. 11.3 are used for deriving a model for the dynamics of the core navigation error variables, and the results are organized into formulas for the respective submatrices $\mathbf{F}_{ij[k]}$, where $[k]$ denotes which of the six navigation calculations is responsible for the dynamic coupling contribution. The complete dynamic coefficient matrix will be the sum of the contributions from all six.

Evaluating the six 3×9 Jacobians requires taking $6 \times 3 \times 9 = 162$ partial derivatives, which were all evaluated in a symbolic mathematics programming environment. Symbolic math programs generally improve efficiency and accuracy, and are recommended for this kind of work.

11.3 NAVIGATION ERROR DYNAMICS

For each of the six suspect calculations, the first-order contributions of the kth contributor to the 9×9 matrix of Eq. 11.13 are calculated as the Jacobian matrix of the respective function $\mathbf{f}[k]$ it performs with respect to the navigation error variables, evaluated at $\boldsymbol{\xi} = 0$. The resulting matrices are used for modeling first-order sensitivities of INS implementation error dynamics to navigation errors:

$$\mathbf{F}_{[k]} \approx \left. \frac{\partial \mathbf{f}_{[k]}}{\partial \boldsymbol{\xi}} \right|_{\xi=0}. \tag{11.14}$$

As mentioned in the paragraph above Eq. 11.13, each function $\mathbf{f}_{[k]}$ will contribute to only one of the three-row block rows of Eq. 11.13. Table 11.3 is a breakdown of that partitioning.

The final dynamic coefficient matrix of the navigation error model will be composed of the sums from each of the six potentially contributing computational processes:

$$\mathbf{F}_{11} = \mathbf{F}_{11[1]} \tag{11.15}$$

$$\mathbf{F}_{12} = \mathbf{F}_{12[1]} \tag{11.16}$$

$$\mathbf{F}_{13} = \mathbf{F}_{13[1]} \tag{11.17}$$

$$\mathbf{F}_{21} = \mathbf{F}_{21[2]} + \mathbf{F}_{21[3]} + \mathbf{F}_{21[4]} \tag{11.18}$$

$$\mathbf{F}_{22} = \mathbf{F}_{22[2]} + \mathbf{F}_{22[3]} + \mathbf{F}_{22[4]} \tag{11.19}$$

$$\mathbf{F}_{23} = \mathbf{F}_{23[2]} + \mathbf{F}_{23[3]} + \mathbf{F}_{23[4]} \tag{11.20}$$

$$\mathbf{F}_{31} = \mathbf{F}_{31[5]} + \mathbf{F}_{31[6]} \tag{11.21}$$

$$\mathbf{F}_{32} = \mathbf{F}_{32[5]} + \mathbf{F}_{32[6]} \tag{11.22}$$

$$\mathbf{F}_{33} = \mathbf{F}_{33[5]} + \mathbf{F}_{33[6]}. \tag{11.23}$$

This is the approach for deriving the first-order error propagation model for the nine core navigation error variables.

11.3.1 Error Dynamics due to Velocity Integration

This is the most straightforward part of the model. The navigation error subvector $\dot{\boldsymbol{\varepsilon}}$ is the error in the INS velocity estimate, and the only modeled error

TABLE 11.3. Dynamic Coefficient Submatrix Sources

Corrupted INS Implementation Processes	Error Coupling Mechanisms		
	Location Error	Velocity Error	Tilt Error
1. Velocity integration	$\mathbf{F}_{1,1[1]}$	$\mathbf{F}_{1,2[1]}$	$\mathbf{F}_{1,3[1]}$
2. Gravity acceleration	$\mathbf{F}_{2,1[2]}$	$\mathbf{F}_{2,2[2]}$	$\mathbf{F}_{2,3[2]}$
3. Coriolis acceleration	$\mathbf{F}_{2,1[3]}$	$\mathbf{F}_{2,2[3]}$	$\mathbf{F}_{2,3[3]}$
4. Centrifugal acceleration	$\mathbf{F}_{3,1[4]}$	$\mathbf{F}_{3,2[4]}$	$\mathbf{F}_{3,3[4]}$
5. Earthrate compensation	$\mathbf{F}_{3,1[5]}$	$\mathbf{F}_{3,2[5]}$	$\mathbf{F}_{3,3[5]}$
6. Leveling	$\mathbf{F}_{3,1[6]}$	$\mathbf{F}_{3,2[6]}$	$\mathbf{F}_{3,3[6]}$

corruption due to velocity integration will the accumulation of navigation location error ε through integration of velocity error $\dot{\varepsilon}$; that is,

$$\frac{d}{dt}\varepsilon = \dot{\varepsilon} \tag{11.24}$$

$$= \begin{bmatrix}\mathbf{F}_{11[1]} & \mathbf{F}_{12[1]} & \mathbf{F}_{13[1]}\end{bmatrix}\boldsymbol{\xi} \tag{11.25}$$

$$\mathbf{F}_{11[1]} = 0\,(3\times 3 \text{ zero matrix}) \tag{11.26}$$

$$\mathbf{F}_{12[1]} = \mathbf{I}\,(3\times 3 \text{ identity matrix}) \tag{11.27}$$

$$\mathbf{F}_{13[1]} = 0\,(3\times 3 \text{ zero matrix}), \tag{11.28}$$

where the "[1]" in the subscript refers to the first error coupling source listed in Table 11.3.

11.3.2 Error Dynamics due to Gravity Calculations

11.3.2.1 INS Gravity Modeling The model used for calculating gravity has to be rather accurate to attain reasonable inertial navigation performance [2]. Geoid models generally limit INS performance to ~10^{-1} nautical mile per hour CEP rate. The INS gravity model will usually include a term of the sort

$$\hat{\mathbf{g}}_{\text{INS}} \approx \begin{bmatrix} \\ 0 \\ 0 \\ \dfrac{-\text{GM}_{\oplus}}{\left[R_{\oplus}\left(\hat{\phi}\right)+\hat{h}\right]^2} \end{bmatrix}$$

$$\hat{R}_{\oplus}\left(\hat{\phi}\right) \approx a_{\oplus}\left(1 - f_{\oplus}\sin^2\hat{\phi}\right),$$

where the variable $R_\oplus(\phi)$ is the reference ellipsoidal geoid surface radius, h is the INS height above that surface, $a_\oplus$ is the semimajor axis of the reference ellipsoid, $f_\oplus$ its flattening, and ϕ the geodetic latitude of the INS.

Geoid models are essentially limited to spherical harmonics of the reference equipotential surface height up to second order. More accurate gravity models include spherical harmonics of a much higher order.[4] Such small-scale variations make little difference to navigation error dynamics, however.

11.3.2.2 Navigation Error Model for Gravity Calculations The dominant term in the dynamic coefficient matrix is the one due to the vertical gradient of gravity. It is the one that makes inertial navigation unstable in the vertical direction.

The other major navigation error propagation effects of gravity miscalculation are caused by misalignments ($\boldsymbol{\rho}$). To capture these effects, the gravity model can be stripped down to a simpler form:

$$\hat{\mathbf{g}}_{\text{INS}} \approx \begin{bmatrix} 0 \\ 0 \\ \dfrac{-\text{GM}_\oplus}{\left[R_\oplus + \varepsilon_{\text{U}}\right]^2} \end{bmatrix}, \tag{11.29}$$

for which the equivalent value in ENU coordinates will be

$$\hat{\mathbf{g}}_{\text{ENU}} \approx C_{\text{ENU}}^{\text{INS}} \hat{\mathbf{g}}_{\text{INS}}. \tag{11.30}$$

The navigation errors in this approximation will cause acceleration errors, which are the time derivatives of $\dot{\boldsymbol{\varepsilon}}$:

$$\frac{d}{dt}\dot{\boldsymbol{\varepsilon}} \approx \left.\frac{\partial \dot{\mathbf{g}}_{\text{ENU}}}{\partial \boldsymbol{\xi}}\right|_{\boldsymbol{\xi}=0} \boldsymbol{\xi} \tag{11.31}$$

$$= \begin{bmatrix} \mathbf{F}_{21[2]} & \mathbf{F}_{22[2]} & \mathbf{F}_{23[2]} \end{bmatrix} \boldsymbol{\xi} \tag{11.32}$$

$$\mathbf{F}_{21[2]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 2\dfrac{\text{GM}_\oplus}{\bar{R}_\oplus^3} \end{bmatrix} \tag{11.33}$$

[4]Up to order 60 in some models.

$$\mathbf{F}_{22[2]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.34}$$

$$\mathbf{F}_{23[2]} = \begin{bmatrix} 0 & -\dfrac{\mathrm{GM}_\oplus}{\bar{R}_\oplus^2} & 0 \\ \dfrac{\mathrm{GM}_\oplus}{\bar{R}_\oplus^2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{11.35}$$

where the "[2]" in the subscript refers to the second error coupling source listed in Table 11.3.

The lone nonzero element in $\mathbf{F}_{21[2]}$ is the cause of vertical error instability, which will be addressed in Section 11.3.9.

11.3.3 Error Dynamics due to Coriolis Acceleration

The calculation for Coriolis correction (derived in Section B.5.3 of Appendix B) is

$$\frac{d}{dt}\mathbf{v}_{\text{Coriolis}} = -2\boldsymbol{\omega}_\oplus \otimes \mathbf{v}. \tag{11.36}$$

However, navigation errors will corrupt its implementation as

$$\frac{d}{dt}\hat{\mathbf{v}}_{\text{Coriolis}} = -2\hat{\boldsymbol{\omega}}_\oplus \otimes (\mathbf{v} + \dot{\boldsymbol{\varepsilon}}) \tag{11.37}$$

$$\hat{\boldsymbol{\omega}}_\oplus = \mathbf{C}_{\text{ENU}}^{\text{INS}} \begin{bmatrix} 0 \\ \cos\hat{\phi}\Omega_\oplus \\ \sin\hat{\phi}\Omega_\oplus \end{bmatrix} \tag{11.38}$$

$$\cos\hat{\phi} \approx \cos\phi - \sin\phi\frac{\varepsilon_{\text{N}}}{\bar{R}_\oplus} \tag{11.39}$$

$$\sin\hat{\phi} \approx \sin\phi + \cos\phi\frac{\varepsilon_{\text{N}}}{\bar{R}_\oplus}, \tag{11.40}$$

the contribution of which to navigation error dynamics will be

$$\frac{d}{dt}\dot{\boldsymbol{\varepsilon}} \approx \left.\frac{\partial \hat{\mathbf{v}}_{\text{Coriolis}}}{\partial \boldsymbol{\xi}}\right|_{\boldsymbol{\xi}=0} \boldsymbol{\xi} \tag{11.41}$$

$$= \begin{bmatrix}\mathbf{F}_{21[3]} & \mathbf{F}_{22[3]} & \mathbf{F}_{23[3]}\end{bmatrix}\boldsymbol{\xi} \tag{11.42}$$

$$\mathbf{F}_{21[3]} = \begin{bmatrix} 0 & 2\dfrac{\Omega_\oplus \sin\phi v_{\text{U}}}{\bar{R}_\oplus} + 2\dfrac{\Omega_\oplus \cos\phi v_{\text{N}}}{\bar{R}_\oplus} & 0 \\ 0 & -2\dfrac{\Omega_\oplus \cos\phi v_{\text{E}}}{\bar{R}_\oplus} & 0 \\ 0 & -2\dfrac{\Omega_\oplus \sin\phi v_{\text{E}}}{\bar{R}_\oplus} & 0 \end{bmatrix} \tag{11.43}$$

$$\mathbf{F}_{22[3]} = \begin{bmatrix} 0 & 2\Omega_\oplus \sin\phi & -2\Omega_\oplus \cos\phi \\ -2\Omega_\oplus \sin\phi & 0 & 0 \\ 2\Omega_\oplus \cos\phi & 0 & 0 \end{bmatrix} \tag{11.44}$$

$$\mathbf{F}_{23[3]} = \begin{bmatrix} 2\Omega_\oplus \sin\phi v_{\text{U}} + 2\Omega_\oplus \cos\phi v_{\text{N}} & 0 & 0 \\ -2\Omega_\oplus \cos\phi v_{\text{E}} & 2\Omega_\oplus \sin\phi v_{\text{U}} & -2\Omega_\oplus \cos\phi v_{\text{U}} \\ -2\Omega_\oplus \sin\phi v_{\text{E}} & -2\Omega_\oplus \sin\phi v_{\text{N}} & 2\Omega_\oplus \cos\phi v_{\text{N}} \end{bmatrix}, \tag{11.45}$$

where the "[3]" in the subscript refers to the third error coupling source listed in Table 11.3.

11.3.4 Error Dynamics due to Centrifugal Acceleration

Centrifugal acceleration is built into the terrestrial gravity model at the surface of the reference geoid. It is the primary reason for its equatorial bulge.[5] As a consequence, the vector sum of gravitational and centrifugal acceleration at sea level is orthogonal to the surface.

The problem is that the respective gradients of centrifugal and gravitational accelerations at the surface are quite different. Gravity decreases as the inverse square of radius, but centrifugal acceleration increases linearly with radius. We have examined how the gravity gradient influences the dynamics of inertial navigation errors, and we need to do the same for centrifugal acceleration. For that purpose, the formula for centrifugal acceleration is derived in the final section of Appendix B. It has the form

[5]A second-order contributor is the earth's mass density redistribution due to centrifugal forces.

$$\hat{\mathbf{a}}_{\text{Centrifugal}} = -[\hat{\boldsymbol{\omega}}_{\oplus} \otimes][\hat{\boldsymbol{\omega}}_{\oplus} \otimes]\hat{\mathbf{x}} \tag{11.46}$$

$$\hat{\mathbf{x}} = \mathbf{x} + \boldsymbol{\varepsilon}, \tag{11.47}$$

where $\mathbf{x}$ is the vector from Earth's center to the INS, and $\hat{\boldsymbol{\omega}}_{\oplus}$ defined by Eq. 11.38. The relevant Jacobian

$$\left.\frac{\partial \hat{\mathbf{a}}_{\text{Centrifugal}}}{\partial \boldsymbol{\xi}}\right|_{\xi=0} = \begin{bmatrix}\mathbf{F}_{21[4]} & \mathbf{F}_{22[4]} & \mathbf{F}_{23[4]}\end{bmatrix} \tag{11.48}$$

$$\mathbf{F}_{21[4]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \Omega_{\oplus}^2 \sin^2 \text{phi} - \Omega_{\oplus}^2 \cos^2 \text{phi} & -\Omega_{\oplus}^2 \sin\phi\cos\phi \\ 0 & -2\Omega_{\oplus}^2 \sin\phi\cos\phi & \Omega_{\oplus}^2 \cos^2 \text{phi} \end{bmatrix} \tag{11.49}$$

$$\mathbf{F}_{22[4]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.50}$$

$$\mathbf{F}_{23[4]} = \begin{bmatrix} 0 & \Omega_{\oplus}^2 \cos^2 \text{phi}\bar{R}_{\oplus} & \Omega_{\oplus}^2 \sin\phi\cos\phi\bar{R}_{\oplus} \\ -\Omega_{\oplus}^2 \cos^2 \text{phi}\bar{R}_{\oplus} & 0 & 0 \\ -\Omega_{\oplus}^2 \sin\phi\cos\phi\bar{R}_{\oplus} & 0 & 0 \end{bmatrix}. \tag{11.51}$$

11.3.5 Error Dynamics due to Earthrate Leveling

Maintaining locally level reference directions on a rotating earth requires that the estimated earth rotation rate be subtracted from the measured rotation rates. The formula for the error in the estimated earth rotation rate vector has already been used in the error analysis of the Coriolis correction, in Eq. 11.38. The Jacobian of the calculated negative earth rotation rate with respect to navigation errors is then

$$\left.\frac{\partial - \hat{\boldsymbol{\omega}}_{\oplus}}{\partial \boldsymbol{\xi}}\right|_{\xi=0} = \begin{bmatrix}\mathbf{F}_{31[5]} & \mathbf{F}_{32[5]} & \mathbf{F}_{33[5]}\end{bmatrix} \tag{11.52}$$

$$\mathbf{F}_{31[5]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \dfrac{\Omega_{\oplus} \sin\phi}{\bar{R}_{\oplus}} & 0 \\ 0 & -\dfrac{\Omega_{\oplus} \cos\phi}{\bar{R}_{\oplus}} & 0 \end{bmatrix} \tag{11.53}$$

$$\mathbf{F}_{32[5]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.54}$$

$$\mathbf{F}_{33[5]} = \begin{bmatrix} 0 & -\Omega_{\oplus}\sin\phi & \Omega_{\oplus}\cos\phi \\ \Omega_{\oplus}\sin\phi & 0 & 0 \\ -\Omega_{\oplus}\cos\phi & 0 & 0 \end{bmatrix}, \tag{11.55}$$

where the "[5]" in the subscript refers to the fourth error coupling source listed in Table 11.3.

11.3.6 Error Dynamics due to Velocity Leveling

Locally level reference directions need to rotate at some vector rate $\boldsymbol{\omega}_v$ as the INS moves with velocity $\boldsymbol{v}$ relative to the curved surface of the earth, as illustrated in Fig. 11.4. The model for this in the navigation implementation can be rather sophisticated because rotation rate depends on the radius of curvature of the reference geoid, and this can be different in different directions and at different locations on the earth. However, this level of rigor is not necessary for modeling the dynamics of navigation errors. A spherical earth model will suffice.

In that case, the operative formula for the locally level coordinate rotation rate as a function of location, velocity, and orientation will be

$$\hat{\boldsymbol{\omega}}_v = \frac{1}{\bar{R}_{\oplus} + \varepsilon_{\mathrm{U}}} \hat{\mathbf{u}}_{\mathrm{U}} \otimes \hat{\mathbf{v}} \tag{11.56}$$

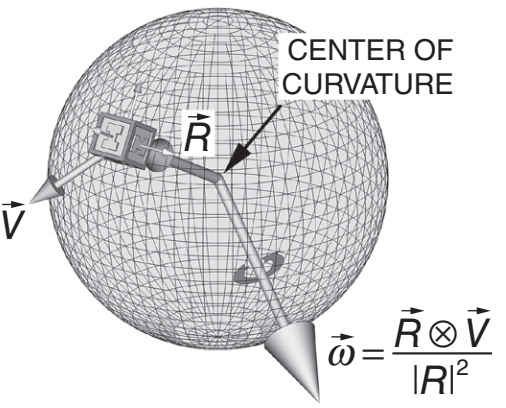


Fig. 11.4 Velocity leveling for terrestrial navigation.

$$= \frac{1}{\bar{R}_\oplus + \varepsilon_U} C_{ENU}^{INS} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \otimes (\mathbf{v} + \hat{\boldsymbol{\varepsilon}}) \tag{11.57}$$

$$\left. \frac{\partial \hat{\boldsymbol{\omega}}_v}{\partial \boldsymbol{\xi}} \right|_{\boldsymbol{\xi}=0} = \begin{bmatrix} \mathbf{F}_{31[6]} & \mathbf{F}_{32[6]} & \mathbf{F}_{33[6]} \end{bmatrix} \boldsymbol{\xi} \tag{11.58}$$

$$\mathbf{F}_{31[6]} = \begin{bmatrix} 0 & 0 & \dfrac{v_N}{\bar{R}_\oplus^2} \\ 0 & 0 & -\dfrac{v_E}{\bar{R}_\oplus^2} \\ 0 & 0 & 0 \end{bmatrix} \tag{11.59}$$

$$\mathbf{F}_{32[6]} = \begin{bmatrix} 0 & -\bar{R}_\oplus^{-1} & 0 \\ \bar{R}_\oplus^{-1} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.60}$$

$$\mathbf{F}_{33[6]} = \begin{bmatrix} -\dfrac{v_U}{\bar{R}_\oplus} & 0 & 0 \\ 0 & -\dfrac{v_U}{\bar{R}_\oplus} & 0 \\ \dfrac{v_E}{\bar{R}_\oplus} & \dfrac{v_N}{\bar{R}_\oplus} & 0 \end{bmatrix}, \tag{11.61}$$

where the "[6]" in the subscript refers to the fifth error coupling source listed in Table 11.3.

11.3.7 Error Dynamics due to Acceleration and Misalignments

INS acceleration in INS coordinates is

$$\mathbf{a}_{INS} = \mathbf{C}_{INS}^{ENU} \mathbf{a}_{ENU} \tag{11.62}$$

$$= \{\mathbf{I} - \boldsymbol{\rho} \otimes\} \mathbf{a}_{ENU} \tag{11.63}$$

$$= \mathbf{a}_{ENU} - \boldsymbol{\rho} \otimes \mathbf{a}_{ENU} \tag{11.64}$$

$$= \mathbf{a}_{ENU} + \mathbf{a}_{ENU} \otimes \boldsymbol{\rho} \tag{11.65}$$

$$= \mathbf{a}_{\mathrm{ENU}} + \begin{bmatrix} 0 & -a_{\mathrm{U}} & a_{\mathrm{N}} \\ a_{\mathrm{U}} & 0 & -a_{\mathrm{E}} \\ -a_{\mathrm{N}} & a_{\mathrm{E}} & 0 \end{bmatrix} \boldsymbol{\rho}, \tag{11.66}$$

where $\mathbf{a}_{\mathrm{ENU}}$ is INS acceleration in ENU coordinates. The vector $\mathbf{a}_{\mathrm{ENU}}$ in this case is not sensed acceleration but the physical acceleration of the INS in ENU coordinates.

The last term of the last equation above represents an acceleration error, which will be integrated by the INS and added to velocity error; that is, the first-order contribution of this term to navigation error dynamics is a term coupling misalignments into the derivative of the velocity error:

$$\mathbf{F}_{23[7]} = \begin{bmatrix} 0 & -a_{\mathrm{U}} & a_{\mathrm{N}} \\ a_{\mathrm{U}} & 0 & -a_{\mathrm{E}} \\ -a_{\mathrm{N}} & a_{\mathrm{E}} & 0 \end{bmatrix}. \tag{11.67}$$

There is a corresponding term for rotation rates coupling misalignments in into the derivative of misalignments. For gimbaled systems, those rotation rates in INS coordinates are those commanded by the INS to remain locally level, and these terms have already been factored into the model as leveling errors.

11.3.8 Composite Model from All Effects

Summing up the contributions from all six INS procedures according to Eqs. 11.15–11.61, the full dynamic coefficient matrix for the core variables will have the submatrices

$$\mathbf{F}_{11} = \mathbf{F}_{11[1]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.68}$$

$$\mathbf{F}_{12} = \mathbf{F}_{12[1]} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{11.69}$$

$$\mathbf{F}_{13} = \mathbf{F}_{13[1]} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.70}$$

$$\mathbf{F}_{21} = \mathbf{F}_{21[2]} + \mathbf{F}_{21[3]} + \mathbf{F}_{21[4]} \tag{11.71}$$

$$= \begin{bmatrix} 0 & 2\dfrac{\Omega_\oplus \sin\phi v_U}{\bar{R}_\oplus} + 2\dfrac{\Omega_\oplus \cos\phi v_N}{\bar{R}_\oplus} & 0 \\ 0 & -2\dfrac{\Omega_\oplus \cos\phi v_E}{\bar{R}_\oplus} + \Omega_\oplus^2 \sin^2 \text{phi} - \Omega_\oplus^2 \cos^2 \text{phi} & -\Omega_\oplus^2 \sin\phi\cos\phi \\ 0 & -2\dfrac{\Omega_\oplus \sin\phi v_E}{\bar{R}_\oplus} - 2\Omega_\oplus^2 \sin\phi\cos\phi & 2\dfrac{GM_\oplus}{\bar{R}_\oplus^3} + \Omega_\oplus^2 \cos^2 \text{phi} \end{bmatrix} \tag{11.72}$$

$$\mathbf{F}_{22} = \mathbf{F}_{22[2]} + \mathbf{F}_{22[3]} + \mathbf{F}_{22[4]} \tag{11.73}$$

$$= \begin{bmatrix} 0 & 2\Omega_\oplus \sin\phi & -2\Omega_\oplus \cos\phi \\ -2\Omega_\oplus \sin\phi & 0 & 0 \\ 2\Omega_\oplus \cos\phi & 0 & 0 \end{bmatrix} \tag{11.74}$$

$$\mathbf{F}_{23} = \mathbf{F}_{23[2]} + \mathbf{F}_{23[3]} + \mathbf{F}_{23[3]} + \mathbf{F}_{23[7]} \tag{11.75}$$

$$= \begin{bmatrix} 2\Omega_\oplus \sin\phi v_U + 2\Omega_\oplus \cos\phi v_N & -a_U - \dfrac{GM_\oplus}{\bar{R}_\oplus^2} + \Omega_\oplus^2 \cos^2 \text{phi}\bar{R}_\oplus & a_N + \Omega_\oplus^2 \sin\phi\cos\phi\bar{R}_\oplus \\ a_U + \dfrac{GM_\oplus}{\bar{R}_\oplus^2} - 2\Omega_\oplus \cos\phi v_E - \Omega_\oplus^2 \cos^2 \text{phi}\bar{R}_\oplus & 2\Omega_\oplus \sin\phi v_U & -a_E - 2\Omega_\oplus \cos\phi v_U \\ -a_N - 2\Omega_\oplus \sin\phi v_E - \Omega_\oplus^2 \sin\phi\cos\phi\bar{R}_\oplus & a_E - 2\Omega_\oplus \sin\phi v_N & 2\Omega_\oplus \cos\phi v_N \end{bmatrix} \tag{11.76}$$

$$\mathbf{F}_{31} = \mathbf{F}_{31[5]} + \mathbf{F}_{31[6]} \tag{11.77}$$

$$= \begin{bmatrix} 0 & 0 & \dfrac{v_N}{\bar{R}_\oplus^2} \\ 0 & \dfrac{\Omega_\oplus \sin\phi}{\bar{R}_\oplus} & -\dfrac{v_E}{\bar{R}_\oplus^2} \\ 0 & -\dfrac{\Omega_\oplus \cos\phi}{\bar{R}_\oplus} & 0 \end{bmatrix} \tag{11.78}$$

$$\mathbf{F}_{32} = \mathbf{F}_{32[5]} + \mathbf{F}_{32[6]} \tag{11.79}$$

$$= \begin{bmatrix} 0 & -\bar{R}_\oplus^{-1} & 0 \\ \bar{R}_\oplus^{-1} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11.80}$$

$$\mathbf{F}_{33} = \mathbf{F}_{33[5]} + \mathbf{F}_{33[6]} \tag{11.81}$$

TABLE 11.4. Equation References for Dynamic Coefficient Submatrices

Corrupted INS Implementation Processes	Error Coupling Mechanisms		
	Location Error	Velocity Error	Tilt Error
1. Velocity integration (equations)	$\mathbf{F}_{1,1[2]}$ (Eq. 11.26)	$\mathbf{F}_{1,2[1]}$ (Eq. 11.27)	$\mathbf{F}_{1,3[1]}$ (Eq. 11.28)
2. Gravity (equations)	$\mathbf{F}_{2,1[2]}$ (Eq. 11.33)	$\mathbf{F}_{2,2[2]}$ (Eq. 11.34)	$\mathbf{F}_{2,3[2]}$ (Eq. 11.35)
3. Coriolis (equations)	$\mathbf{F}_{2,1[3]}$ (Eq. 11.43)	$\mathbf{F}_{2,2[3]}$ (Eq. 11.44)	$\mathbf{F}_{2,3[3]}$ (Eq. 11.44)
4. Centrifugal (equations)	$\mathbf{F}_{2,1[4]}$ (Eq. 11.49)	$\mathbf{F}_{2,2[4]}$ (Eq. 11.50)	$\mathbf{F}_{2,4[3]}$ (Eq. 11.50)
5. Earthrate compensation (equations)	$\mathbf{F}_{3,1[5]}$ (Eq. 11.53)	$\mathbf{F}_{3,2[5]}$ (Eq. 11.54)	$\mathbf{F}_{3,3[5]}$ (Eq. 11.55)
6. Leveling (equations)	$\mathbf{F}_{3,1[6]}$ (Eq. 11.59)	$\mathbf{F}_{3,2[6]}$ (Eq. 11.60)	$\mathbf{F}_{3,3[6]}$ (Eq. 11.61)

$$= \begin{bmatrix} -\dfrac{v_U}{\bar{R}_\oplus} & -\Omega_\oplus \sin\phi & \Omega_\oplus \cos\phi \\ \Omega_\oplus \sin\phi & -\dfrac{v_U}{\bar{R}_\oplus} & 0 \\ -\Omega_\oplus \cos\phi + \dfrac{v_E}{\bar{R}_\oplus} & \dfrac{v_N}{\bar{R}_\oplus} & 0 \end{bmatrix}, \tag{11.82}$$

and the corresponding equation numbers for the component submatrices $\mathbf{F}_{i,j}[k]$ are listed in Table 11.4.

The full 9×9 dynamic coefficient matrix for navigation errors, in 3×3 block form, is then

$$\mathbf{F}_{\text{core}} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & \mathbf{F}_{13} \\ \mathbf{F}_{21} & \mathbf{F}_{22} & \mathbf{F}_{23} \\ \mathbf{F}_{31} & \mathbf{F}_{32} & \mathbf{F}_{33} \end{bmatrix}, \tag{11.83}$$

where the corresponding values of the 3×3 submatices are defined above.

These equations are implemented to form $\mathbf{F}_{\text{core}}$ in the MATLAB® m-file `Fcore9.m` on the accompanying website.

11.3.9 Vertical Navigation Instability

The term "vertical channel" is sometimes used for the vertical components of location and velocity in inertial navigation. The vertical channel is the "dark

side" of inertial navigation, because stand-alone inertial navigation errors in the vertical channel are naturally unstable.

The lower-right matrix element in Eq. 11.33 is the source of vertical navigation instability. Its value is positive, meaning that upward acceleration error *increases* with upward location (altitude) error:

$$\frac{d}{dt}\dot{\varepsilon}_{\mathrm{U}} = 2\frac{\mathrm{GM}_{\oplus}}{\bar{R}_{\oplus}^{3}}\varepsilon_{\mathrm{U}}, \tag{11.84}$$

or the equivalent state space form for navigation errors in the vertical channel,

$$\frac{d}{dt}\begin{bmatrix}\varepsilon_{\mathrm{U}}\\ \dot{\varepsilon}_{\mathrm{U}}\end{bmatrix} = \begin{bmatrix}0 & 1\\ \tau_{\mathrm{U}}^{-2} & 0\end{bmatrix}\begin{bmatrix}\varepsilon_{\mathrm{U}}\\ \dot{\varepsilon}_{\mathrm{U}}\end{bmatrix} \tag{11.85}$$

$$\tau_{\mathrm{U}} = \sqrt{\frac{\bar{R}_{\oplus}^{3}}{2\mathrm{GM}_{\oplus}}} \tag{11.86}$$

$$\approx 570 \text{ s}, \tag{11.87}$$

$$\approx 9.5 \text{ min}. \tag{11.88}$$

The general solution to this linear time-invariant system, as a function of time $t > t_0$ and initial conditions at $t = t_0$, is

$$\begin{bmatrix}\varepsilon_{\mathrm{U}}(t)\\ \dot{\varepsilon}_{\mathrm{U}}(t)\end{bmatrix} = \underbrace{\begin{bmatrix}\cosh\left(\dfrac{t-t_0}{\tau_{\mathrm{U}}}\right) & \tau_{\mathrm{U}}\sinh\left(\dfrac{t-t_0}{\tau_{\mathrm{U}}}\right)\\ \sinh\left(\dfrac{t-t_0}{\tau_{\mathrm{U}}}\right)\tau_{\mathrm{U}}^{-1} & \cosh\left(\dfrac{t-t_0}{\tau_{\mathrm{U}}}\right)\end{bmatrix}}_{\boldsymbol{\Phi}}\begin{bmatrix}\varepsilon_{\mathrm{U}}(t_0)\\ \dot{\varepsilon}_{\mathrm{U}}(t_0)\end{bmatrix}. \tag{11.89}$$

This solution diverges exponentially with time, as shown in Fig. 11.5. This is a plot of the elements of the state transition matrix $\boldsymbol{\Phi}$ over a period of 10 h, during which the solution grows by a factor approaching Avogadro's number ($\approx 6.0221415 \times 10^{23}$).

Inertial navigation for ballistic missiles must include vertical navigation because that is the direction of most of the action. However, an INS for ballistic missiles typically has extremely small initial navigation errors; it spends long periods of time in self-calibration before it is launched; and its total period of navigation is in the order of several minutes. Fortunately, all these factors reduce the impact of vertical navigation instability on performance to levels that can be tolerated.

Vertical navigation instability is not a problem for ships at sea either because they have no need to navigate in the vertical direction.

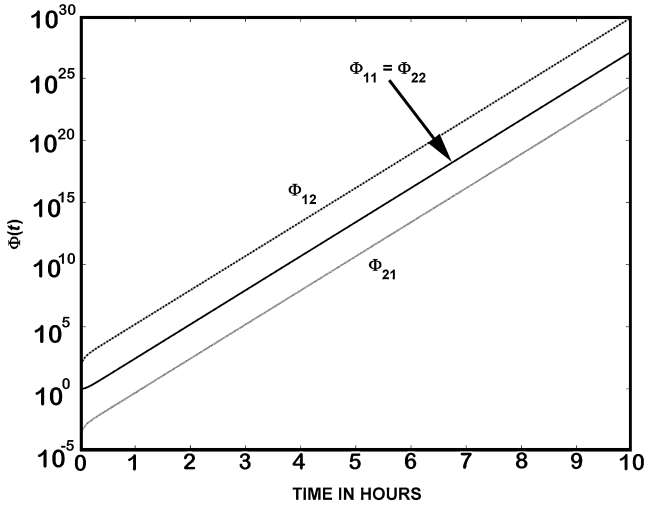


Fig. 11.5 Growth of vertical channel errors over 10 hours.

It could have been a problem for aircraft navigation, except that aircraft already had reasonably reliable vertical information from altimeters. They would be able to navigate more like a ship but use their altimeters for vertical navigation. They could also use altimeters for "aiding" a full three-dimensional INS, in much the same way it is done in GNSS/INS integration. The latter approach led to augmentation of the navigation error model to include error characteristics of the altimeter, and eventually to methods for recalibrating critical sensors for even better performance.

11.3.9.1 Altimeter Aiding It is done using barometric altimeters, which have biases with long-term drift due to ambient barometric pressure variations in the dynamic atmosphere. These variations have different statistics at different altitudes and different parts of the world, but they generally have correlation times in the order of a day for most of the continental United States [3]. The RMS variations are in the order of a few millibars, roughly equivalent to 10^2 m in altitude. The resulting altimeter bias is usually modeled as an exponentially correlated random process with a correlation time constant of ~1 day and an RMS value of ~10^2 m. In that case, the dynamic coefficient matrix for the augmented system will have the block form

$$\mathbf{F}_{10} = \begin{bmatrix} \mathbf{F}_9 & 0 \\ 0 & -1/\tau_{\text{alt}} \end{bmatrix}, \tag{11.90}$$

where τ_{alt} is the correlation time for altimeter bias errors.

A modeled demonstration of altimeter damping of an INS, using this model, is implemented in the MATLAB® m-file `Fcore10Test1.m` on the

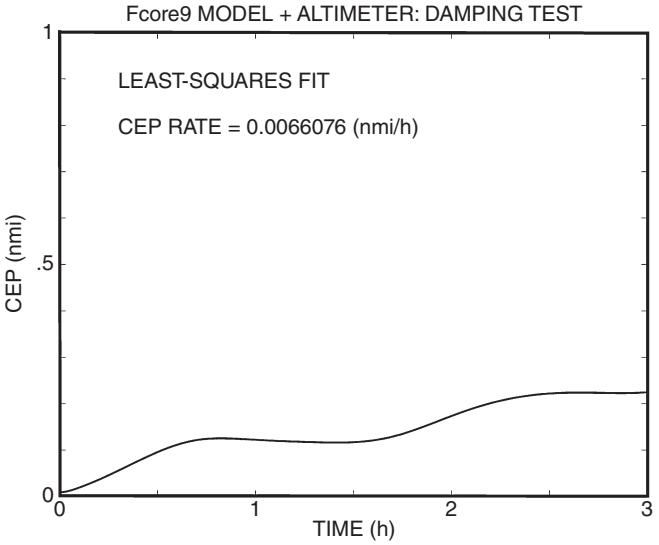


Fig. 11.6 CEP plot from m-file `F10CEPrate.m`.

accompanying website. This script solves the Riccati equation for this system with specified initial uncertainties and plots the resulting RMS uncertainties in all 10 state variables. It also plots the resulting CEP as a function of time and performs a least-squares fit to estimate CEP rate. The output for the latter is shown in Fig. 11.6. The estimated CEP rate is ~0.07 nmi/h. This may seem small, but it is for a system in which the only process noise is from the altimeter model, and the only source of sensor noise is on the altimeter output. The results do appear to indicate that the presence of even small altitude errors does corrupt system dynamics, even without error sources other than those due to the altimeter.

Example 11.1 Three-State Model for Altimeter Aiding. Exponentially correlated processes have dynamic models of the sort

$$\dot{\varepsilon}_{\text{alt}} = -\frac{\varepsilon_{\text{alt}}}{\tau_{\text{alt}}} + w_{\text{alt}}(t), \tag{11.91}$$

where τ_{alt} is the exponential correlation time of the altimeter errors, $w_{\text{alt}}(t)$ is a zero-mean white-noise process with variance

$$q_{\text{alt}} = \frac{2\sigma_{\text{alt}}^2}{\tau_{\text{alt}}}, \tag{11.92}$$

and σ_{alt}^2 is the mean-squared altimeter error.

If this altimeter were used as an auxiliary sensor for measuring altitude, then the three-state (altitude, altitude rate, and altimeter bias) dynamic model for just the vertical channel with altimeter aiding would be

$$\frac{d}{dt}\begin{bmatrix}\varepsilon_{\mathrm{U}}\\ \dot{\varepsilon}_{\mathrm{U}}\\ \varepsilon_{\mathrm{alt}}\end{bmatrix}=\underbrace{\begin{bmatrix}0 & 1 & 0\\ \tau_{\mathrm{U}}^{-2} & 0 & 0\\ 0 & 0 & \dfrac{-1}{\tau_{\mathrm{alt}}}\end{bmatrix}}_{\mathbf{F}_{\mathrm{alt}}}\begin{bmatrix}\varepsilon_{\mathrm{U}}\\ \dot{\varepsilon}_{\mathrm{U}}\\ \varepsilon_{\mathrm{alt}}\end{bmatrix}+\begin{bmatrix}0\\ w_{\mathrm{U}}(t)\\ w_{\mathrm{alt}}(t)\end{bmatrix}, \tag{11.93}$$

where $\mathbf{F}_{\mathrm{alt}}$ is the 3×3 dynamic coefficient matrix for this system.

The altimeter output is the sum of real altitude and the slowly varying altimeter bias, so the associated measurement sensitivity matrix for the altimeter output has the form

$$\mathbf{H}_{\mathrm{alt}}=[1 \quad 0 \quad 1]. \tag{11.94}$$

Whether the covariance matrix $\mathbf{P}_{\infty}$ of mean-squared vertical channel uncertainties settles to a finite value with this setup is determined by whether or not the steady-state matrix Riccati equation for this system

$$0=\mathbf{F}_{\mathrm{alt}}\mathbf{P}_{\infty}+\mathbf{P}_{\infty}\mathbf{F}_{\mathrm{alt}}^{T}-\frac{\mathbf{P}_{\infty}\mathbf{H}_{\mathrm{alt}}^{T}\mathbf{H}_{\mathrm{alt}}\mathbf{P}_{\infty}}{R_{\mathrm{alt}}}+\begin{bmatrix}0 & 0 & 0\\ 0 & q_{\mathrm{U}} & 0\\ 0 & 0 & q_{\mathrm{alt}}\end{bmatrix} \tag{11.95}$$

has a solution. In this representation, R_{alt} is the mean-squared altimeter noise, q_{U} is the covariance of vertical accelerometer noise, and $\mathbf{P}_{\infty}$ is the steady-state covariance of state estimation uncertainty (if it exists). This Riccati equation is equivalent to six independent quadratic equations constraining the six independent elements of $\mathbf{P}_{\infty}$. The solution can be determined numerically, however, and it is finite.

Example 11.2 10-State INS Error Model for Altimeter Aiding. The simulation result shown in Fig. 11.6 was generated using a 10-state model with the altimeter bias as the 10th state variable. The measurement sensitivity matrix for the altimeter will be the 1×10 matrix,

$$\mathbf{H}_{\mathrm{alt}}=[0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1], \tag{11.96}$$

modeling the fact that the altimeter measures the vertical position error, as well as its own bias, and some values for the altimeter bias time-constant and

mean-squared altimeter error due to ambient atmospheric pressure variations. In this example,

$$\sigma_{\text{alt}} = 100 \text{ m, the effect of ambient pressure variation,} \tag{11.97}$$

$$\tau_{\text{alt}} = 1 \text{ day correlation time for ambient atmospheric pressure variation,} \tag{11.98}$$

with sampling every 10 s. The atmospheric pressure standard deviations and correlation time are from Ref. 3. The values chosen are in the midrange of Klein's hemisphere-wide values but are reasonable for the United States.

The MATLAB® m-file `DampingTest10part1.m` on the accompanying website runs a covariance analysis of a stationary INS with altimeter damping. The m-file prints the histories of all ten RMS state variable uncertainties, including the one shown in Fig. 11.6. This shows the damping reduction of initial RMS altitude uncertainty by a factor or around four, whereas the undamped RMS value would have increased by several orders of magnitude.

This test is only the beginning of the verification process for the Fcore9 model, which requires some means of holding vertical channel errors in check while the rest of the INS error model is verified. This simple case includes no inertial sensor noise.

11.3.10 Schuler Oscillations

In 1906, the German gyrocompass inventor Hermann Anschütz-Kaempfe invited his cousin, Maximilian Schuler (1882–1972), to look into why Anschütz-Kaempfe's gyrocompasses were not providing reliable bearing information on ships encountering high sea states. In analyzing the physics of the problem, Schuler determined that the problem had to do with torques induced by lateral accelerations, and the best solution would be to design a pendulous suspension for the gyrocompass with a period of about 84 min, that being the period of an ideal pendulum (i.e., with massless support arm) with support arm length equal to the radius of the earth, $R_\oplus$ [6]. The dynamic equation for small displacements δ of a pendulum with that support arm length in the near-earth gravitational field is

$$\begin{aligned}\frac{d^2}{dt^2}\delta &\approx \frac{-g}{R_\oplus}\delta \\ &= -\frac{\text{GM}_\oplus}{R_\oplus^3}\delta \\ &= -\omega_{\text{Schuler}}^2\delta\end{aligned}$$

$$\begin{aligned}\omega_{\text{Schuler}} &= \sqrt{\frac{GM_{\oplus}}{R_{\oplus}^3}} \\ &\approx \sqrt{\frac{0.3986004\times 10^{15}}{\left(0.6371009\times 10^{7}\right)^3}} \\ &\approx 0.001241528\ (\text{rad/s}) \\ f_{\text{Schuler}} &\approx 0.0001975953\ (\text{Hz}),\end{aligned}$$

$$\begin{aligned}T_{\text{Schuler}} &= 1/f_{\text{Schuler}} \\ &\approx 1/0.0001975953 \\ &\approx 84\ \text{min},\end{aligned}$$

called the *Schuler period.* It was rediscovered with the advent of inertial navigation, a consequence of gravity modeling on the earth with radius $R_{\oplus}$, just like the Schuler pendulum. As a consequence, due to the Coriolis effect, horizontal INS errors tend to behave like an ideal Schuler/Foucault pendulum.

Figure 11.7 shows a surface plot of the trajectory of INS position errors, generated using the MATLAB® INS toolbox from GPSoft, starting with an initial north velocity error. It clearly shows the Schuler oscillations, and the turning of the plane of oscillation due to Coriolis acceleration. This is the sort

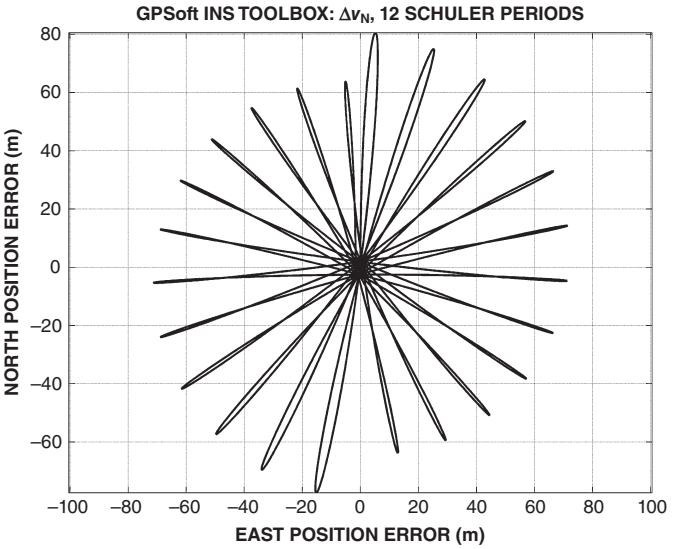


Fig. 11.7 Schuler–Coriolis demonstration using the MATLAB® INS toolbox from GPSoft.

of model behavior that must be compared to the response of the actual INS to the same initial error conditions.

11.3.11 Core Model Validation and Tuning

All models need to be verified by comparing modeled behavior with the behavior of the system they were designed to represent. It is often the case that the INS and its model evolve together so that each validates the other as development progresses.

After the INS and its model are ready to be tested, the first step in INS/model verification is usually to deliberately introduce navigation errors into the INS and its model and to compare the respective responses. As a minimum, this involves nine tests for the INS and its model: one for each error variable.

Model "tuning" is the modification of model parameters to better represent the behavior of the modeled system. These modifications are usually related to sensor error characteristics, which have not yet been addressed.

The model derived above has not been vetted in this manner, but it will be used to demonstrate the verification procedure. Any model should be trusted only after a verification procedure with the INS it is designed to represent.

11.3.11.1 Horizontal Inertial Navigation Model Ships afloat do not need to navigate vertically. They still need three gyroscopes, but for gimbaled navigation, they can eliminate the vertical accelerometer and the calculations that would otherwise doubly integrate its outputs.

For that type of INS, one can also eliminate altitude and altitude rate from the core nine-state model, eliminating two state variables. The result is the dynamic coefficient matrix implemented in the MATLAB® m-file `Fcore7.m` on the accompanying website. It was obtained by removing the third and sixth rows and columns of the dynamic coefficient matrix from the 10-state model, and by setting vertical velocity to zero. Seven-state models can be used for predicting INS response to deliberate initialization errors in each of the seven state variables. These seven test conditions are implemented in the MATLAB® m-files `Fcore7Test1.m` through `Fcore7Test7.m` on the accompanying website. These plot the modeled INS error response to initial errors in easting, northing, east velocity, north velocity, east tilt, north tilt, and heading. The dominating response in most cases is Schuler oscillation with the plane of motion rotated by the Coriolis effect, similar to that shown in Fig. 11.7.

11.4 INERTIAL SENSOR NOISE

A major source of dynamic process noise driving navigation error distributions is from noise in the gyroscopes and accelerometers, described in Section 3.3 of Chapter 3.

Zero-mean white sensor noise (described in Section 3.3.1) does not change the dynamic model structure of inertial navigation errors, except by adding a noise process $\mathbf{w}(t)$:

$$\frac{d}{dt}\boldsymbol{\xi} = \mathbf{F}_{\text{core}}\boldsymbol{\xi} + \mathbf{w}(t). \tag{11.99}$$

The white noise from the accelerometers is integrated into velocity error $\dot{\boldsymbol{\varepsilon}}$, and white noise from the gyroscopes is integrated into orientation error. This results in a process noise covariance with covariance structure

$$\mathbf{Q}_{\text{sensor noise}} = E\left\langle \mathbf{w}(t)\mathbf{w}^T(t)\right\rangle \tag{11.100}$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & \mathbf{Q}_{\text{acc.}} & 0 \\ 0 & 0 & \mathbf{Q}_{\text{gyro}} \end{bmatrix}, \tag{11.101}$$

where $\mathbf{Q}_{\text{acc.}}$ is the accelerometer noise covariance and $\mathbf{Q}_{\text{grro}}$ is the gyro noise covariance. Unless there is data confirming noise correlation between sensors, these submatrices may be scalar matrices:

$$\mathbf{Q}_{\text{acc.}} = q_{\text{acc.}}\mathbf{I} \tag{11.102}$$

$$\mathbf{Q}_{\text{gyro}} = q_{\text{gyro}}\mathbf{I}, \tag{11.103}$$

where the accelerometer noise variance $q_{\text{acc.}}$ is the same for all accelerometers, and gyro noise variance q_{gyro} is the same for all gyros. However, empirical values of $\mathbf{Q}_{\text{acc.}}$ and $\mathbf{Q}_{\text{gyro}}$ can be determined by computing the actual 3×3 covariances from all three accelerometers and from all three gyroscopes.

The propagation equation for the covariance matrix $\mathbf{P}_{\xi}$ of navigation errors in this case has either of the forms

$$\frac{d}{dt}\mathbf{P}_{\xi} = \mathbf{F}_{\text{core}}\mathbf{P}_{\xi} + \mathbf{P}_{\xi}\mathbf{F}_{\text{core}}^T + \mathbf{Q}_{\text{sensor noise}} \tag{11.104}$$

$$\mathbf{P}_{\xi[k]} = \boldsymbol{\Phi}_{\xi[k-1]}\mathbf{P}_{\xi[k-1]}\boldsymbol{\Phi}_{\xi[k-1]}^T + \Delta t\mathbf{Q}_{\text{sensor noise}} \tag{11.105}$$

$$\boldsymbol{\Phi}_{\xi[k-1]} = \exp\left[\mathbf{F}_{\text{core}}\left(\hat{\boldsymbol{\xi}}(t_{k-1}), \Delta t\right)\right]. \tag{11.106}$$

in either continuous time or discrete time. These equations characterize the expected performance of an INS with given sensor noise, in terms of how fast the mean-squared navigation errors can be expected to deteriorate over time.

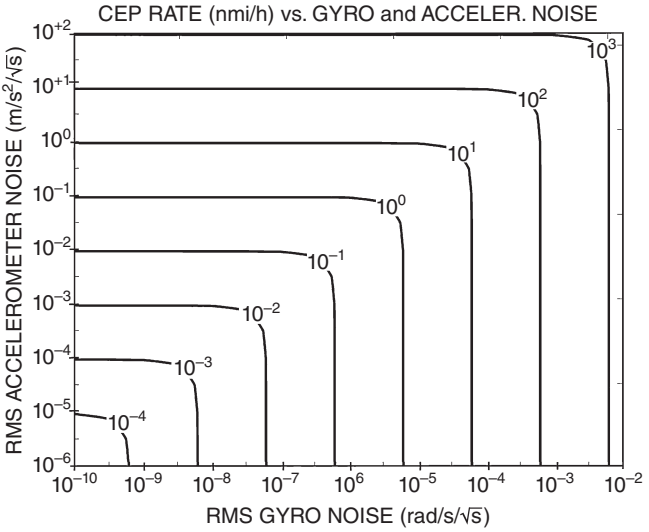


Fig. 11.8 Estimated CEP rate versus gyro and accelerometer noise.

11.4.1 CEP Rate versus Sensor Noise

The m-file `CEPvsSensorNoise.m` on the accompanying website uses Eq. 11.105 to obtain least-squares estimates of CEP rate over a range of gyro and accelerometer noise levels for horizontal inertial navigation, with only sensor noise and no sensor compensation errors. The results, plotted in Fig. 11.8, would indicate that gyro noise dominates INS performance if RMS gyro noise (in radian per second per root-second) is greater than ~10^{-4} times RMS accelerometer noise (in meters per second squared per root-second), and accelerometer noise dominates INS performance if RMS accelerometer noise is greater than ~10^4 times RMS gyro noise. However, *these results do not include the effects of vertical navigation errors.*

11.5 SENSOR COMPENSATION ERRORS

As a rule, the extreme accuracy requirements for sensors in inertial navigation cannot be achieved through manufacturing precision alone. The last few bits of accuracy are generally achieved using sensor calibration to characterize the residual error pattern so that these errors can be adequately compensated during navigation. The models and methods used for this purpose are described in Section 3.4.

It is not uncommon for the values of sensor compensation parameters to change a little between system turn-ons, and even to change during operation after turn-ons. The phenomenon has been reduced in some instances by design

changes but has not been totally eliminated. As an alternative, many schemes have been used for partial recalibration of the more suspect compensation parameters. These have generally relied on using additional sensors. With the arrival of GNSS, much attention has been focused on using essentially the same approach for INS recalibration using GNSS. The necessary models are developed in this chapter. The specific application using GNSS is the subject of the next chapter.

11.5.1 Sensor Compensation Error Models

The sensor compensation models used here are the same as those used in Chapter 3, except the compensation parameters are no longer necessarily constant. Attention is focused primarily on those compensation parameters deemed more likely to drift. These are primarily the zero-order (bias) and first-order (scale factor an input axis misalignment) compensation parameters. Input axis misalignments tend to be relatively stable, so the usual suspects are scale factors and biases.

The notation will be changed to reflect the fact that we are dealing with variables, not constants. The variations due to drift in the bias, scale factor, and input axis misalignment for the accelerometers and gyroscopes in the general case can then be modeled as

$$\boldsymbol{\delta}_a(t) = \begin{bmatrix} \delta_{ab\mathrm{E}} \\ \delta_{ab\mathrm{N}} \\ \delta_{ab\mathrm{U}} \end{bmatrix} + \mathbf{M}_{\mu a}\mathbf{a}_{\mathrm{comp.0}} \tag{11.107}$$

$$\boldsymbol{\delta}_\omega(t) = \begin{bmatrix} \delta_{gb\mathrm{E}} \\ \delta_{gb\mathrm{N}} \\ \delta_{gb\mathrm{U}} \end{bmatrix} + \mathbf{M}_{\mu g}\boldsymbol{\omega}_{\mathrm{comp.0}} \tag{11.108}$$

$$\mathbf{M}_{a\mu} = \begin{bmatrix} \delta_{asE} & \delta_{a\mu\mathrm{EN}} & \delta_{a\mu\mathrm{EU}} \\ \delta_{a\mu\mathrm{NE}} & \delta_{as\mathrm{N}} & \delta_{a\mu\mathrm{NU}} \\ \delta_{a\mu\mathrm{UE}} & \delta_{a\mu\mathrm{UN}} & \delta_{as\mathrm{U}} \end{bmatrix} \tag{11.109}$$

$$\mathbf{M}_{g\mu} = \begin{bmatrix} \delta_{gs\mathrm{E}} & \delta_{g\mu\mathrm{EN}} & \delta_{g\mu\mathrm{EU}} \\ \delta_{g\mu\mathrm{NE}} & \delta_{gs\mathrm{N}} & \delta_{g\mu\mathrm{NU}} \\ \delta_{g\mu\mathrm{UE}} & \delta_{g\mu\mathrm{UN}} & \delta_{gs\mathrm{U}} \end{bmatrix}, \tag{11.110}$$

where

$\mathbf{a}_{\mathrm{comp.0}}$ is the vector of accelerometer outputs compensated using the prior values of the compensation parameters;

$\boldsymbol{\omega}_{\text{comp.0}}$ is the vector of gyro outputs compensated using the prior values of the compensation parameters;

$\boldsymbol{\delta}_a(\text{t})$ is the error in compensated sensed acceleration due to compensation parameter drift;

$\boldsymbol{\delta}_\omega(\text{t})$ is the error in compensated sensed rotation rate due to compensation parameter drift;

$\delta_{\text{ab}*}$ is the accelerometer bias drift (with * = E, N, or U);

$\delta_{\text{gb}*}$ is the gyro bias drift (with * = E, N, or U);

$\mathbf{M}_{a\mu}$ is the drift in the accelerometer scale factor and input axis misalignment matrix; and

$\mathbf{M}_{g\mu}$ is the drift in the gyro scale factor and input axis misalignment matrix.

With the assumption that the drift in input axis misalignments is insignificant, the model for each sensor type (accelerometer or gyroscope) has only six drifting parameters: three biases and three scale factors. The models for the errors in the sensed quantities then have only diagonal entries in the scale factor an input axis misalignment matrices:

$$\mathbf{M}_{a\mu} = \begin{bmatrix} \delta_{\text{asE}} & 0 & 0 \\ 0 & \delta_{\text{asN}} & 0 \\ 0 & 0 & \delta_{\text{asU}} \end{bmatrix} \tag{11.111}$$

$$\mathbf{M}_{g\mu} = \begin{bmatrix} \delta_{\text{gsE}} & 0 & 0 \\ 0 & \delta_{\text{gsN}} & 0 \\ 0 & 0 & \delta_{\text{gsU}} \end{bmatrix}. \tag{11.112}$$

11.5.1.1 Exponentially Correlated Parameter Drift Model Models for slowly varying sensor compensation errors are usually random walk models or exponentially correlated process models. The latter are preferred because they have finite bounds on their variances, and they allow the analyst to use model parameters (i.e., RMS error and error correlation time) based on test data.

The dynamic model for an independent exponentially correlated random process $\{\eta(t)\}$ has the general form

$$\frac{d}{dt}\eta = \frac{-1}{\tau}(\eta - \bar{\eta}) + w(t), \tag{11.113}$$

where $\bar{\eta}$ is the long-term mean of the process, τ is the process correlation time, and the white process noise $w(t) \in N(0, 2\sigma^2/\tau)$, where $\sigma^2 = \left\langle (\eta - \bar{\eta})^2 \right\rangle$ is the mean-squared variation in the sensor compensation parameter. In the case of

parameter drift, the long-term average drift can usually be assumed to be zero, in which case $\bar{\delta} = 0$ and the model becomes

$$\frac{d}{dt}\delta = \frac{-1}{\tau}\delta + w(t), \tag{11.114}$$

with the same parameters.

Equation 11.114 is already in the form of linear stochastic equation. If 12 of these (for scale factor and bias drift of six sensors) were to be added to the dynamic system model for an inertial system, it would add a 12×12 scalar matrix to the dynamic coefficient matrix,

$$\mathbf{F}_{\text{drift}} = -\tau^{-1}\mathbf{I}, \tag{11.115}$$

assuming all parameters have the same correlation times. If not, each element on the diagonal of $\mathbf{F}_{\text{drift}}$ can have a different negative value.

The corresponding process noise covariances may be different for accelerometers and gyroscopes, and different for bias and scale factor for each type of sensor. In general, the process noise covariance may have a 12×12 diagonal structure with 3×3 diagonal blocks:

$$\mathbf{Q}_{\text{drift}} = \begin{bmatrix} \dfrac{2\sigma_{\text{ab}}^2}{\tau_{\text{ab}}}\mathbf{I} & 0 & 0 & 0 \\ 0 & \dfrac{2\sigma_{\text{as}}^2}{\tau_{\text{as}}}\mathbf{I} & 0 & 0 \\ 0 & 0 & \dfrac{2\sigma_{\text{gb}}^2}{\tau_{\text{gb}}}\mathbf{I} & 0 \\ 0 & 0 & 0 & \dfrac{2\sigma_{\text{gs}}^2}{\tau_{\text{gs}}}\mathbf{I} \end{bmatrix}, \tag{11.116}$$

where, for the case in which all sensors of each type have the same parameter drift parameters but statistically independent distributions,

σ_{ab}^2, τ_{ab} are the variance and correlation time, respectively, of accelerometer biases;

σ_{as}^2, τ_{as} are the variance and correlation time, respectively, of accelerometer scale factors;

σ_{gb}^2, τ_{gb} are the variance and correlation time, respectively, of gyro biases;

σ_{gs}^2, τ_{gs} are the variance and correlation time, respectively, of gyro scale factors; and

$\mathbf{I}$ is a 3×3 identity matrix.

However, it is generally possible—and recommended—that the process noise parameters for compensation errors be determined from actual sensor error data under operating conditions. There is always the possibility that sensor compensation error drift is driven by common-mode random processes such as temperature or power-level variation, and that information can be exploited to make a more useful model.

11.5.1.2 Dynamic Coupling into Navigation Errors There is no first-order dynamic coupling of navigation errors into sensor errors, but sensor errors do couple directly into the time derivatives of navigation errors. The general formulas for navigation error dynamics due to compensation errors are

$$\frac{d}{dt}\begin{bmatrix}\dot{\varepsilon}_{\mathrm{E}}\\ \dot{\varepsilon}_{\mathrm{N}}\\ \dot{\varepsilon}_{\mathrm{U}}\end{bmatrix} = \boldsymbol{\delta}_a(t) \tag{11.117}$$

$$= \begin{bmatrix}\delta_{\mathrm{abE}}\\ \delta_{\mathrm{abN}}\\ \delta_{\mathrm{abU}}\end{bmatrix} + \mathbf{M}_{\mu a} a_{\mathrm{comp.0}} \tag{11.118}$$

$$\frac{d}{dt}\begin{bmatrix}\rho_{\mathrm{E}}\\ \rho_{\mathrm{N}}\\ \rho_{\mathrm{U}}\end{bmatrix} = \boldsymbol{\delta}_\omega(t) \tag{11.119}$$

$$= \begin{bmatrix}\delta_{\mathrm{gbE}}\\ \delta_{\mathrm{gbN}}\\ \delta_{\mathrm{gbU}}\end{bmatrix} + \mathbf{M}_{\mu g} \omega_{\mathrm{comp.0}}, \tag{11.120}$$

where the six affected INS navigation errors are

$\dot{\varepsilon}_{\mathrm{E}}$, $\dot{\varepsilon}_{\mathrm{N}}$, and $\dot{\varepsilon}_{\mathrm{U}}$, the INS velocity errors;
ρ_{E}, ρ_{N}, and ρ_{U}, the INS tilt and heading errors;

and the matrices $\mathbf{M}_{\mu a}$ and $\mathbf{M}_{\mu g}$ are defined by Eqs. 11.111 and 11.112.

11.5.1.3 Augmented Dynamic Coefficient Matrix Let the order of the 12 state variables in the state vector for sensor compensation errors be

$$\boldsymbol{\delta} = \begin{bmatrix}\boldsymbol{\delta}_a\\ \boldsymbol{\delta}_\omega\end{bmatrix} \tag{11.121}$$

$$\boldsymbol{\delta}_a = \begin{bmatrix}\boldsymbol{\delta}_{\mathrm{ab}}\\ \boldsymbol{\delta}_{\mathrm{as}}\end{bmatrix} \quad \boldsymbol{\delta}_\omega = \begin{bmatrix}\boldsymbol{\delta}_{\mathrm{gb}}\\ \boldsymbol{\delta}_{\mathrm{gs}}\end{bmatrix} \tag{11.122}$$

$$\boldsymbol{\delta}_{\text{ab}} = \begin{bmatrix} \boldsymbol{\delta}_{\text{abE}} \\ \boldsymbol{\delta}_{\text{abN}} \\ \boldsymbol{\delta}_{\text{abU}} \end{bmatrix}, \quad \boldsymbol{\delta}_{\text{as}} = \begin{bmatrix} \boldsymbol{\delta}_{\text{asE}} \\ \boldsymbol{\delta}_{\text{asN}} \\ \boldsymbol{\delta}_{\text{asU}} \end{bmatrix} \quad \boldsymbol{\delta}_{\text{gb}} = \begin{bmatrix} \boldsymbol{\delta}_{\text{gbE}} \\ \boldsymbol{\delta}_{\text{gbN}} \\ \boldsymbol{\delta}_{\text{gbU}} \end{bmatrix} \quad \boldsymbol{\delta}_{\text{gs}} = \begin{bmatrix} \boldsymbol{\delta}_{\text{gsE}} \\ \boldsymbol{\delta}_{\text{gsN}} \\ \boldsymbol{\delta}_{\text{gsU}} \end{bmatrix}, \tag{11.123}$$

then the augmented dynamic coefficient matrix for a system including an altimeter would have the block form

$$\mathbf{F}_{\text{aug}} = \begin{bmatrix} \mathbf{F}_{\text{core}} & 0 & \mathbf{F}_{\delta\to\xi} \\ 0 & -\tau_{\text{alt}}^{-1} & 0 \\ 0 & 0 & \mathbf{F}_{\text{drift}} \end{bmatrix}, \tag{11.124}$$

where the 9×12 submatrix coupling compensation parameter errors into navigation errors is

$$\mathbf{F}_{\delta\to\xi} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \mathbf{I} & \mathbf{D}_a & 0 & 0 \\ 0 & 0 & \mathbf{I} & \mathbf{D}_\omega \end{bmatrix} \tag{11.125}$$

$$\mathbf{D}_a = \begin{bmatrix} a_{\text{E}} & 0 & 0 \\ 0 & a_{\text{N}} & 0 \\ 0 & 0 & a_{\text{U}} \end{bmatrix} \tag{11.126}$$

$$\mathbf{D}_\omega = \begin{bmatrix} \omega_{\text{E}} & 0 & 0 \\ 0 & \omega_{\text{N}} & 0 \\ 0 & 0 & \omega_{\text{U}} \end{bmatrix}, \tag{11.127}$$

with diagonal entries equal to input acceleration and rotation rate components, respectively.

The corresponding process noise covariance for the resulting 9 + 1 + 12 = 22-state INS error model would then be

$$\mathbf{Q}_{\text{INS}} = \begin{bmatrix} \mathbf{Q}_{\text{sensor noise}} & 0 & 0 \\ 0 & q_{\text{alt}} & 0 \\ 0 & 0 & \mathbf{Q}_{\text{drift}} \end{bmatrix}, \tag{11.128}$$

where $\mathbf{Q}_{\text{sensor noise}}$ is defined by Eq. 11.101, q_{alt} is defined by Eq. 11.92, and $\mathbf{Q}_{\text{drift}}$ is defined by Eq. 11.116.

At this point, q_{alt} is essentially a placeholder for what will become GNSS signal delay covariance in Chapter 12. This model requires something to stabilize vertical navigation errors.

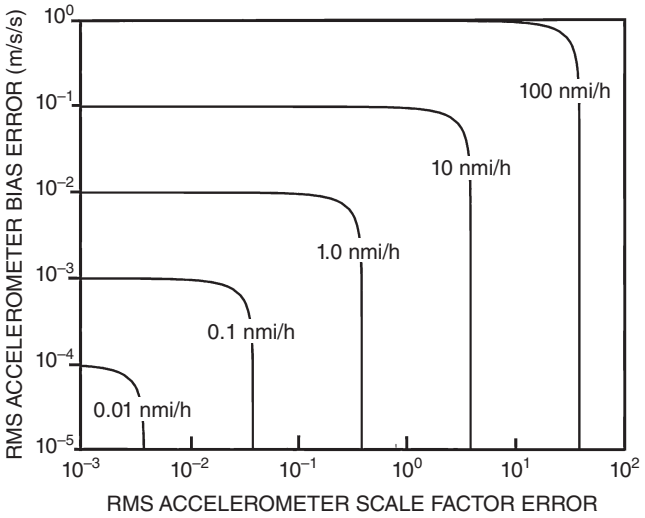


Fig. 11.9 CEP slope versus accelerometer compensation drift parameters, based on test track similation.

The alternative 20-state model for horizontal inertial navigation has the same structure, with the third and sixth rows and columns of **F** and **Q** deleted.

The MATLAB® m-file `CEPvsAccComp.m` on the accompanying website analyzes the influence of accelerometer bias and scale factor drift on INS CEP rates, assuming no other error sources. CEP rate is evaluated for 2500 sets of sensor compensation drift rates, using 6 h of simulation in each case. Compensation error is modeled as an exponentially correlated random process with a correlation time of 1 h. The analysis uses simulated test track conditions to provide sensor inputs to drive the navigation errors. Otherwise, sensitivity to accelerometer scale factor errors would not be detectable. Using a closed track for the test conditions makes the average acceleration equal to zero, which casts some aspersions on the applicability of the results to other test conditions. Performance is evaluated using two different estimates of CEP rate: one simply making a straight-line fit through the origin and the other computing independent slope and intercept values. Also, the simulation is for a gimbaled system, which is not sensitive to gyro scale factor errors. Results showing straight-line-fit CEP rates as a function of RMS accelerometer compensation errors are shown in Fig. 11.9.

11.6 SOFTWARE SOURCES

Software developed for this book is intended for instructional purposes only. It has not been vetted against any inertial system and is not recommended for

commercial purposes. As alternatives, Mathworks distributes MATLAB® toolboxes for INS and GPS analysis from GPSoft, and there are other potential sources listed on the World Wide Web.

11.7 SUMMARY

1. If inertial navigation is "navigation in a box," then whatever happens in that box stays in that box. This includes any navigation errors that cannot be sensed from inside the box, such as errors in the INS estimates of location, velocity, and orientation.
2. Inertial systems are ordinarily initialized with the starting values of location, velocity, and orientation, but—without the use of auxiliary sensors—there is no way of detecting errors in the navigation solution thereafter.
3. The "navigation solution" for an INS includes estimates of location, velocity, and orientation. These estimates are propagated forward in time, using only the inputs from gyroscopes and accelerometers.
4. INS analysis is the study of how errors in the navigation solution behave over time and how this behavior depends on the error characteristics of the sensors and the dynamics of the intended mission applications.
5. This kind of analysis is a fundamental part of the development cycle for inertial navigation systems. It provides a reliable model for judging how the selection of sensors influences system performance, for verifying INS software implementations, and for diagnosing test results.
6. Although modeling accuracy is extremely important in the implementation of inertial navigation, the same is not true for error analysis. Error variables typically represent relative errors in the order of parts per thousand or less. As a consequence, first-order approximations are generally adequate for INS error analysis. Although matrix operations in covariance analysis of inertial navigation systems may require good numerical precision to maintain stability of the results, the analysis results are not overly sensitive to the accuracy of the models used in the INS implementation.
7. However, the first-order-only modeling used here is not exact and is not intended to represent all inertial sensors or all INS implementations. Dynamic effects deemed insignificant for the theoretical sensor have been ignored. Before deciding what is insignificant, it is always necessary to validate the model by comparing its behavior to that of the INS it is intended to represent, using the dynamic conditions the INS is designed for. The models derived here have not been vetted in this manner for any hardware implementation.

8. INSs measure three components of acceleration and three components of rotation, and estimate three components each representing the velocity, location, and orientation of the inertial sensor assembly (ISA). The primary navigational error propagation model for inertial navigation also has three components each, representing the velocity, location, and orientation of the ISA—nine state variables in total. Models for the dynamics of these state variables are derived.
9. Inertial navigation in the terrestrial environment exhibits horizontal location error oscillations with a period of about 84 min. This is called *Schuler oscillation*, and it is due to the shape of the gravitational field. This is not necessarily a bad thing, however. The change in the modeled inertial direction of gravitational acceleration with horizontal displacement results in an effective restoring force on the location error, which forces the location error back toward zero. If it were not for this effect, inertial navigation errors would be worse.
10. The same gravitational field shape causes instability of vertical errors in inertial navigation. Fortunately, an altimeter solves the problem quite well at most altitudes.
11. Navigation in earth-fixed coordinates also introduces error dynamics due to the Coriolis effect. That and Schuler oscillation can make horizontal location errors follow a trajectory similar to that of a Foucault pendulum.
12. The dynamic model for navigation errors can be augmented to include sensor noise, and this model can be used to predict how INS performance depends on sensor noise.
13. The result is a model for how sensor compensation errors influence navigation errors over time. These model derivations are limited to zeroth-order errors (sensor biases) and first-order errors (scale factors and input axis misalignments). The sensor models required for any particular INS application will be determined by the error characteristics of the sensors to be used and on the dynamic conditions they are likely to experience on the intended mission trajectories.
14. The influence on INS performance of "aiding" with other sensors is also a part of INS analysis. An analytical model was used for assessing the influence of an altimeter on vertical navigation. The next chapter will cover INS aiding using GNSS.

Additional background material on inertial navigation systems analysis can be found in the 1971 textbook by Britting [1], and—if you can find it—a 1973 Intermetrics report by Widnall and Grundy [8]. There are no comparable comprehensive treatments for strapdown systems, but some error models can be found in Refs. 5 and 7. However, even these models may not cover all sensors and all inertial navigation implementations. Every sensor and implementation must be used to validate its intended model.

PROBLEMS

11.1 Derive a calibration model for a sensor with third-order errors; that is, its output z_{out} has both affine (linear plus offset) and cubic dependency on its input z_{in}:

$$z_{\text{out}} = c_0 + c_1 z_{\text{in}} + c_3 z_{\text{in}}^3.$$

Given n samples of input–output pairs $\{[z_{k,\text{in}}, z_{k,\text{out}}] | k = 1, 2, \ldots, n\}$, what is the least-squares estimate of the sensor bias c_0, scale factor c_1 and cubic coefficient c_3?

11.2 What is the minimum number of input–output pairs required for solving the problem above? Is it just the number of samples that determines observability? What, exactly, does determine observability for least-squares problems?

11.3 What is the compensation model corresponding to the error model

$$z_{\text{out}} = c_0 + c_1 z_{\text{in}} + c_3 z_{\text{in}}^3?$$

That is, what is the formula for z_{in}, given z_{out}, c_0, c_1, and c_3?

11.4 Derive the first-order compensation error model for the third-order model above, by taking the partial derivatives of the compensated sensor output with respect to the coefficients c_0 (bias), c_1 (scale factor), and c_3 (cubic error coefficient).

11.5 Make a copy of the m-file `CEPvsSensorNoise.m` under another name so that it can be modified. Modify the renamed m-file to include the 10-state model rather than the 7-state model. Run it to obtain the equivalent to Fig. 11.8, but with vertical channel dynamics dampened by a barometric altimeter. Compare the results with those in Fig. 11.8.

11.6 Replace the sensor noise model in the previous problem with a model for sensor compensation errors so that sensor errors are ignored but are replaced by sensor compensation errors. Use the equations in this chapter for the dynamic coefficient matrix (Eq. 11.124) and process noise covariance matrix (Eq. 11.116). Assume correlation times of an hour for all parameters and find what combinations of compensation error covariances will yield CEP rates in the order of 1 nmi/h. Assume that the corresponding process noise covariances for gyro parameters are 10^{-4} times the corresponding parameters for accelerometers, and vary the relative covariances of biases and scale factors to find the ratios at which dominance shifts between bias errors and scale factor errors.

11.7 Run the m-file `CEPvsAccComp.m`, which estimates CEP rates as a function of accelerometer compensation parameter drift. It also plots the radial error as a function of time for the case that no inertial navigation is used at all, with the estimated position remaining at the starting point. How would you compare the other results with the no-navigation case? What might this say about basing INS performance on one set of test conditions?

REFERENCES

[1] K. R. Britting, *Inertial Navigation Systems Analysis*. Artech House, Norwood, MA, 2010.

[2] R. M. Edwards, "Gravity Model Performance in Inertial Navigation," *Journal of Guidance, Control, and Dynamics* **5**, 73–78 (1982).

[3] W. H. Klein, "A Hemispheric Study of Daily Pressure Variability at Sea Level and aloft," *Journal of Meteorology* **8**, 332–346 (1951).

[4] L. A. McGee and S. F. Schmidt, Discovery of the Kalman Filter as a Practical Tool for Aerospace and Industry, National Aeronautics and Space Administration report NASA-TM-86847, 1985.

[5] P. G. Savage, *Introduction to Strapdown Inertial Navigation Systems*, Vols. 1 & 2. Strapdown Associates, Maple Plain, MN, 1996.

[6] M. Schuler, "Die Störung von Pendul-und Kreiselapparaten durch die Beschleunigung der Fahrzeuges," *Physikalische Zeitschrift* **B**, 24 (1923).

[7] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*. IEE, London, 2004.

[8] W. S. Widnall and P. A. Grundy, Inertial Navigation System Error Model, Report TR-03-73, Intermetrics Inc., Cambridge, MA, 1973.

12

GNSS/INS INTEGRATION

Nature laughs at the difficulties of integration.
—Pierre Simon de Laplace (1749–1827)

12.1 CHAPTER FOCUS

12.1.1 Objective

In order to make the best use of its combined resources, global navigation satellite system (GNSS)/inertial navigation system (INS) integration requires fundamental changes in how both navigation modes are jointly implemented. Both subsystems can now use a unified model for what they are doing and work on the navigation problem together. Key features of this approach include the following:

1. The GNSS receiver no longer needs a stochastic model for the dynamics of its host vehicle. It is now integrated with another system that measures those dynamics directly.
2. The GNSS receiver no longer needs to track the frequency and phase changes in its source signals by detecting its own tracking error. It is now integrated with another system that measures those host vehicle accelerations and gyrations which had been causing those errors.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

3. The INS had been accumulating measured accelerations and attitude rates to keep track of its own position and orientation, but had also accumulated some error in the process. Formerly, that error had corrupted its own reckoning of the unsensed accelerations it must account for, and this had compounded matters.
4. The INS is now partnered with a sensor system that measures position directly. The INS is finally able to escape the confines of its Newtonian paradigm and to come out of its box. It is now able to use the GNSS receiver measurements to estimate and correct its own navigation, and its sensor compensation errors. This results in better overall navigation performance whenever GNSS signals are either unavailable or otherwise fail to provide adequate position observability.
5. The integrated system is still passive and stealthy, although it does depend on GNSS signal availability. However, when the integrated system is forced to rely on inertial sensors only, it is now able to do a better job of inertial sensor error compensation.
6. Improved estimates of position and velocity from the INS-only solution make GNSS signal detection and acquisition much easier and faster.

This chapter is about that unified model for GNSS/INS navigation and how it is implemented and evaluated. Its essential elements related to the INS were developed in the previous two chapters. The essential elements related to GNSS were developed in Chapters 4–7.

12.1.2 Order of Presentation

1. Overview of GNSS/INS integration approaches, including a brief historical background
2. Overview of the fundamental integration model:
   (a) the unified navigation model
   (b) where the parts come from
   (c) how they fit together to form a complete GNSS/INS navigation model
3. Implementation of the navigation solution and its self-evaluation equations
4. Performance evaluations on simulated mission trajectories

12.2 GNSS/INS INTEGRATION OVERVIEW

12.2.1 Historical Background

Early successes in integrated inertial navigation were so cloaked in secrecy that very little is known about it to this day. What is clear, however, is that

development of global satellite navigation by the United States Department of Defense was driven by the need for integrated inertial navigation.

Within a few days after the October 4, 1957 launch of the Sputnik I satellite by the Soviet Union, Drs. William H. Guier and George C. Weiffenbach, two scientists at the Applied Physics Laboratory (APL) of Johns Hopkins University, determined that they could estimate the orbital parameters of the satellite by monitoring the Doppler shifts of the transmitted frequency signal from a known geodetic location. Furthermore, it was determined that the inverse problem could be solved by using known satellite orbital parameters and a single set of horizon-to-horizon Doppler shift data to determine the location of the monitoring station to an accuracy of less than 100 m (CEP). With help from Frank McClure at APL, the team submitted a proposal to the Navy's then-concurrent SSBN Polaris Program in 1958 to precisely determine the locations of these nuclear-powered, ballistic missile-carrying submarines using radio navigation sources on satellites. The SSBN fleet could operate undetected for months, but needed to update their inertially derived positions before Polaris missile launches. The first successful tests of the resulting Navy TRANSIT satellite concept were made in 1960.

There was a period of competing triservice space navigation concepts within the Department of Defense in the late 1960s, from about 1964 to 1968. This included the Navy Improved TRANSIT Program, a Naval Research Laboratory TIMATION experiment using side-tone ranging for trilateration, and more precise space-qualified quartz or atomic clocks. These early projects focused on military imperatives for passive global navigation using signals from space but neglected such military requirements as resistance to jamming and denial of use to adversaries. The Aerospace Corporation worked with the Air Force to focus on these issues and on the need for improved real-time three-dimensional navigation for tactical aircraft. Requirements for precise position and velocity during high-speed maneuvering with conventional weapons during all hours and weather conditions would require an order of magnitude improvement over what was available at the time. This effort would become the Air Force 621B Program, which eventually led to the Global Positioning System (GPS). Even the Army joined in the space funding battles and launched satellite ranging transponders for their Sequential Correlation of Range (SECOR) concept. In 1968, the Joint Chiefs of Staff attempted to adjudicate the interservice rivalry through an Executive Steering Group, which then directed extensive classified studies that came to be called "program paralysis by analysis." Much of this work is still classified or lost in the archives, and many of the key investigators are no longer with us.

From 1971 to 1974, the Aerospace Corporation 621B Program developed and tested a spread-spectrum ranging system with a prototype Magnavox Research Laboratory receiver at the White Sands Missile Range, using simulated synchronized emitters on the desert floor and in a balloon. Aircraft postflight position accuracy was demonstrated to be better than 5 m, and velocity errors less than 0.3 m/s. For a handful of those involved, these results

clearly indicated the potential for synthesizing a truly global and sufficiently secure and precise satellite ranging signal that could be optimally combined with current INSs to provide robust guidance navigation to meet current and many future needs for military navigation.

One of those visionaries is Edward H, Martin,[1] who provided much of the historical information presented here. Martin had developed inertial navigation integration methods at Autonetics Research Engineering Division during the late 1960s, when he worked with Dr. Thomas W. DeVries on implementing extended Kalman filters for updating the Autonetics N-16 INS navigation solution using various navigation aids. North American Aviation, the parent company of Autonetics, merged with Rockwell Standard in 1967 to become what would be Rockwell International. By that time, Martin had developed a real-time discrete Kalman filter for integrating Doppler radar into the inertial navigation solution for the F-111/FB-111 MKII avionics.

In 1972, Martin joined Magnavox as Program Development Manager for Military Navigation and helped form the Magnavox/Intermetrics team that, along with General Dynamics, won the 1974 Phase I Concept Development contract for what would become GPS/INS integrated navigation. Directed by Air Force Captain Mel Birnbaum, this team developed seven separate integration algorithms and demonstrated precise blind bombing results by 1977 with loosely coupled INS integration. One of the Magnavox lead software analysts, Anthony Abbott, eventually returned to the Aerospace Corporation and collaborated with John Lukesh of Northrup to pioneer the new high-speed, tightly coupled, B-2 GPS/INS/SAR (synthetic aperture radar) guidance mechanization. So much of this effort was classified that there is little public knowledge of it, and these early successes of GPS/INS integration have largely gone unnoticed. Another largely forgotten bit of history is that it was the need for INS integration that drove the initial development of GNSS.

12.2.2 The Loose/Tight Ranking

Although a unified GNSS/INS navigation model was known at the beginning of GPS development, there were good reasons for not starting with the first attempt. One reason was to avoid the risks and complexity of making significant alterations to functioning stand-alone GPS receivers and inertial navigators, but a more compelling reason may have been to avoid risks associated with restructuring the signal processing hardware and software. Also, available flight-qualified processors at the time had rather limited processing throughput and much slower data transfer rates than we have grown accustomed to. For the first proof-of-concept testing, a “looser” form of GPS/INS integration was chosen.

[1]Member of the GPS development team awarded the 1992 Robert J. Collier Trophy by the National Aeronautics Association and winner of the 2009 Captain P.V.H. Weems Award, presented by the Institute of Navigation, for his role in GPS receiver development.

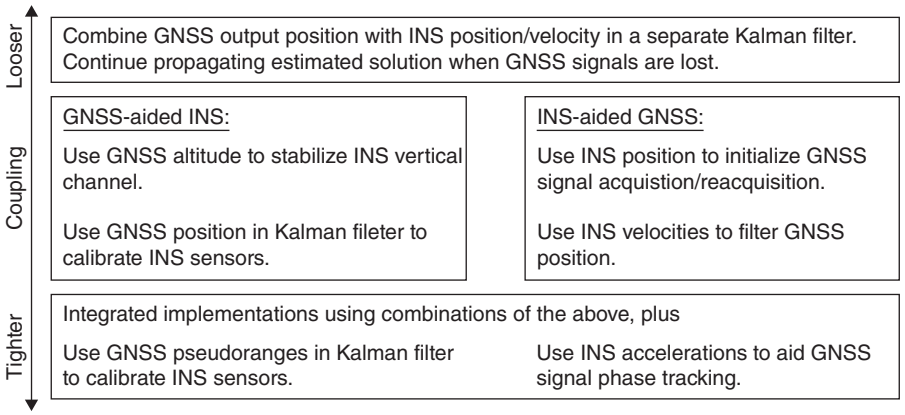


Fig. 12.1 Loosely and tightly coupled implementations.

Figure 12.1 shows a ranking of approaches to GNSS/INS integration, based on how certain implementations functions are altered, and ranked according to the degree of coupling between the otherwise independent implementations of GNSS receivers and inertial navigators. The rankings are loosely called "loose" and "tight," according to the degree to which the independent implementations are modified.

12.2.2.1 Loosely Coupled Implementations The most loosely coupled implementation uses only the standard navigation solutions from the GNSS receiver and INS as inputs to a filter (a Kalman filter, usually), the output of which is the combined estimate of the navigation solution. Although each subsystem (GNSS or INS) may already include its own Kalman filter, this "ultraloosely coupled" integration architecture does not necessarily modify those.

The down side of this approach is that neither the GNSS solution nor the INS solution is made any better by it. As the INS estimate gets worse over time, its filter weighting gets smaller. Eventually, it is essentially ignored, and the integrated solution is the GNSS solution. In situations with inadequate GNSS signal coverage, the INS is of no use.

12.2.2.2 More Tightly Coupled Implementations As a rule, the more tightly coupled implementations have greater impact on the internal implementations of the GNSS receiver and the INS but have better overall performance.

For example, the more tightly coupled implementations may use nonstandard subsystem outputs. Raw pseudoranges from the GNSS receiver may be used independently of the navigation solution they produce. Similarly, raw accelerations from the inertial navigator may be used independently of the

navigation solution they produce. These outputs generally require software changes within both the GNSS receiver and the INS, and it may even require hardware changes to make such data available for the combined solution.

In the most tightly coupled implementations, all data are available for a unified navigation solution. In that case, the unified filter model used for system integration must include variables such as GNSS signal propagation delays, or accelerometer bias and scale factor errors. The estimated values of these variables may be used in the internal implementations of the GNSS receiver and INS.

12.2.2.3 Ultratightly Coupled Integration This term is applied to GNSS/INS integration in which signal tracking loops in the GNSS receiver are augmented by using accelerations and rotations sensed by the INS. This approach can reduce phase-tracking filter lags and improve phase locking during periods of high maneuvering, reduced signal strength, or signal jamming. It also improves GNSS signal acquisition times by reducing Doppler shift uncertainties. This effectively improves navigation accuracy by reducing these navigation error sources and provides additional operational margins for extreme dynamics, weak GNSS signals, or interference from other signals—including jamming. For a discussion of this approach using Kalman filtering, see Ref. 1.

12.2.2.4 Limitations This loose/tight ordering is not "complete," in the sense that it is not always possible to decide whether one implementation is strictly looser or tighter than another. There are just too many degrees of freedom in making implementation changes for them to be ordered in one dimension. However, the ranking does provide some notion of what is involved.

12.2.3 Unified Navigation Model

In this approach, the navigation solution is the one determined by the integrated GNSS/INS implementation. Key features of this approach are the following:

1. The implementation uses a Kalman filter to combine all sensor measurements, taking into account the measurement noise and other error variables associated with each error type. This would include all the navigation error variables of the INS error model, plus all the error variables of GNSS navigation.
2. The combined navigation solution is updated by measurements of all types (pseudoranges, sensed accelerations, and inertial attitude rates).
3. In the short term, between GNSS solutions, the combined navigation solution is updated from INS sensor measurements. These updates use only sensed accelerations and sensed attitude rates, as compensated by

the current estimates of sensor compensation parameters. Updates of the covariance matrix of state estimation uncertainty during these periods reflect changes due to these measurements, as well as the effects of inertial sensor noise. The covariance update is the most computationally intensive part of a Kalman filter, but it does not necessarily have to be run each time the inertial sensor outputs are sampled. Sampling intervals generally range from milliseconds (for most strapdown systems) to tens of milliseconds (for most gimbaled systems).

4. Each GNSS pseudorange measurement is used to update the navigation solution, as well as observable variables of the error model. These observable variables include the signal propagation delays and inertial sensor compensation parameters, which are not observable from the INS measurements. The intersample intervals for GNSS pseudoranges are typically in the order of a second or more for each satellite.
5. The complete navigation solution includes the locations and velocities of the INS and of the GNSS receiver antenna, which cannot be colocated as a rule. The estimated antenna velocity can be used for computing the estimated Doppler velocity on each satellite signal, which aids in signal phase and frequency tracking. Also, the acceleration of the antenna can be determined from the measured INS rotation rates and accelerations, which helps to reduce signal phase and frequency tracking error.
6. The solution also includes updates of the sensor compensation parameters, and updates of location and velocity with estimated INS navigation errors removed.
7. Performance is determined by the solution of the Riccati equations, which is an integral part of the Kalman filter implementation. However, the model used for solving the Riccati equation does not necessarily need to be as precise as the models used for the rest of the navigation solution.

The unified model approach uses a common model for all the dynamic variables required for GNSS/INS integration. It represents the best possible navigation solution from GNSS/INS integration, given all the measurements and their associated uncertainties.

A unified model for integrated GNSS/INS navigation would include everything that contributes to navigation error of the integrated system, including all the error sources and error dynamics, and especially any variables that can be estimated from the combined measurements of the GNSS receiver and the INS.

This would include, as a minimum, state variables for all nine navigation error variables (seven if for horizontal navigation only), all significant inertial sensor compensation errors, variables representing the receiver clock error (bias and drift, as a minimum), and variables representing unknown signal propagation delays due to atmospheric effects.

12.3 UNIFIED MODEL FOR GNSS/INS INTEGRATION

The unified model includes terms from all sensor subsystems, including GNSS and INS. The model used for errors in the GNSS receiver clock is described next.

12.3.1 GNSS Error Models

12.3.1.1 Receiver Clock Error Model A GNSS receiver clock model must be included in the analysis because errors in the receiver clock must also be taken into account when assessing performance.

GNSS receiver clocks generally keep "GPS time," or the equivalent time for other GNSS systems. These are all referenced to Universal Time, Coordinated (UTC), but without the leap seconds. However, clock error models used in GNSS analysis generally use distance units for clock bias, and velocity units for clock drift rates. This is because the clock is being used for timing wave fronts, which are traveling at the speed of light, and the distance traveled is the variable of interest. The equivalent variables and parameters in the GNSS receiver clock model are obtained by multiplying the usual time units by the speed of light.

The GNSS receiver clock model used in the analysis has two state variables:

$\delta_{\text{clock bias}}$, clock bias (in meters), and
$\delta_{\text{clock drift}}$, clock drift rate (in meters per second).

Clock drift rate is assumed to be exponentially correlated. Its stochastic dynamic model has the form

$$\frac{d}{dt}\begin{bmatrix}\delta_{\text{clock bias}}\\ \delta_{\text{clock drift}}\end{bmatrix}=\begin{bmatrix}0 & 1\\ 0 & -1/\tau_{\text{clock drift}}\end{bmatrix}\begin{bmatrix}\delta_{\text{clock bias}}\\ \delta_{\text{clock drift}}\end{bmatrix}+\begin{bmatrix}0\\ w_{\text{clock drift}}(t)\end{bmatrix}, \tag{12.1}$$

with noise variance

$$q_{\text{clock drift}} \overset{\text{def}}{=} \underset{t}{E}\left\langle w^2_{\text{clock drift}}(t)\right\rangle \tag{12.2}$$

$$= \frac{2\sigma^2_{\text{clock drift}}}{\tau_{\text{clock drift}}}, \tag{12.3}$$

where $\sigma^2_{\text{clock drift}}$ is the steady-state variance of clock drift.

This model does not include short-term phase-flicker noise, which is assumed to be filtered out by the clock electronics.

Values used in the analysis assume RMS drift rates of 10 m/s^2 and a correlation time of an hour; that is,

$$\sigma_{\text{clock drift}} \approx 10\ (\text{m/s}) \tag{12.4}$$

$$\tau_{\text{clock drift}} \approx 3600\ (\text{s}) \tag{12.5}$$

$$q_{\text{clock drift}} \approx 0.05\ (\text{m}^2/\text{s}^3). \tag{12.6}$$

In a "cold start" simulation without prior satellite tracking, the initial value for RMS drift rate uncertainty would be its steady-state value:

$$\sigma_{\text{clock drift}}(t_0) = \sigma_{\text{clock drift}}(\infty) \tag{12.7}$$

$$= 10\ (\text{m/s}) \tag{12.8}$$

$$\sigma_{\text{clock bias}}(\infty) = \infty; \tag{12.9}$$

that is, there is no corresponding finite steady-state value for clock bias uncertainty with this model. Fortunately, bootstrap satellite signal acquisition requires significantly less timing accuracy than navigation. Before satellite signal acquisition, the receive clock time is used primarily for selecting candidate satellites from stored ephemerides, which does not require accurate timing. Signal acquisition requires less clock frequency accuracy than navigation, and the clock bias correction is usually initialized as part of acquisition. In navigation analysis, its RMS initial uncertainty is usually set closer to its steady-state tracking value, which is in the order of a few meters, equivalent to an RMS receiver timing error of ~10^{-8} s.

The MATLAB® function ClockModel2 on the accompanying website calculates the values of the dynamic coefficient matrix, process noise covariance matrix, pseudorange sensitivity matrix, and initial covariance matrix value as a function of $\sigma_{\text{dock drift}}$ and $\tau_{\text{dock drift}}$.

12.3.1.2 Atmospheric Propagation Delay Model Perhaps the greatest single contributor to location error in single-frequency GNSS receivers is the residual propagation delay error due to seemingly random variations in atmospheric impedance at GNSS carrier frequencies. The resulting signal arrival delays cause errors in the estimated pseudoranges from the satellite antennas to the receiver antenna.

Propagation delay errors are modeled in distance units, in much the same way as clock errors. These are modeled as independent, exponentially correlated errors, in which case the associated dynamic coefficient matrix and

[2]This is comparable to the performance of the quartz oscillator in a Timex wristwatch with ambient indoor temperatures [2].

process noise covariance matrix are diagonal matrices of the same dimension:

$$\mathbf{F}_{\text{sat. del.}} = \frac{-1}{\tau_{\text{sat. del.}}} \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \tag{12.10}$$

$$\mathbf{Q}_{\text{sat. del.}} = q_{\text{sat. del.}} \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}. \tag{12.11}$$

12.3.1.3 Pseudorange Measurement Noise The measurements are pseudoranges, but the measured time delay values will generally contain some short-term additive noise in the nanosecond range due to electronic noise and signal processing noise. This is modeled as a measurement noise covariance matrix. This will be a diagonal matrix, unless there are significant common-mode error sources such as additive power supply noise or grounding noise.

12.3.2 INS Error Models

12.3.2.1 Navigation Error Model The dynamic coefficient matrix has already been derived in Chapter 11, Eq. 11.77.

12.3.2.2 Sensor Compensation Errors Changes in input axis misalignments are not usually significant enough to be included in the model. That eliminates 12 variables. Also, because the model is for a gimbaled system, errors in gyro scale factors can be ignored. The resulting model has only nine error state variables. Because compensation parameter drift is modeled as an exponentially correlated process, the resulting dynamic coefficient matrix and process noise covariance matrix are diagonal.

The dynamic coupling of compensation errors into navigation errors, given in Eq. 11.119, will be missing its final three rows and columns:

$$\mathbf{F}_{\delta\to\xi} = \begin{bmatrix} 0 & 0 & 0 \\ \mathbf{I} & \mathbf{D}_a & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix} \tag{12.12}$$

$$\mathbf{D}_a = \begin{bmatrix} a_{\mathrm{E}} & 0 & 0 \\ 0 & a_{\mathrm{N}} & 0 \\ 0 & 0 & a_{\mathrm{U}} \end{bmatrix}. \tag{12.13}$$

12.3.3 GNSS/INS Error Model

12.3.3.1 State Variables So far, the error models for GNSS and INS include the following variables that can potentially be estimated as part of the generalized navigation solution:

1. GNSS error sources
   (a) Receiver clock error (two variables)
   (b) Satellite signal delay errors (=number of satellites[3])
2. INS error sources

   (a) Errors in the navigation solution
      i. Position errors (three components)
      ii. Velocity errors (three components)
      iii. Orientation errors (three components)
   (b) Inertial sensor errors
      i. Accelerometer errors
         A. Output bias errors (three components, total)
         B. Scale factor errors (three components, total)
         C. Input axis misalignment errors (six components, total)
      ii. Gyroscope errors
         A. Output bias errors (three components, total)
         B. Scale factor errors (three components, total)
         C. Input axis misalignment errors (six components, total)

The grand total, if all of these were in the model, would be 66 state variables in an integrated GNSS/INS error model. However, there are reasons for leaving some of these out of the model.

12.3.3.2 Numbers of State Variables There are, in total, 9 navigation error variables, 24 first-order sensor compensation parameters, and 2 clock model variables. Assuming around a dozen satellites may be used at any one time, the potential number of state variables for an integrated system would be around 9 + 24 + 2 + 12 = 47. However, not all of these variables need to be

[3]Thirty-one in the ephemerides of July 7, 2012, used in the simulations.

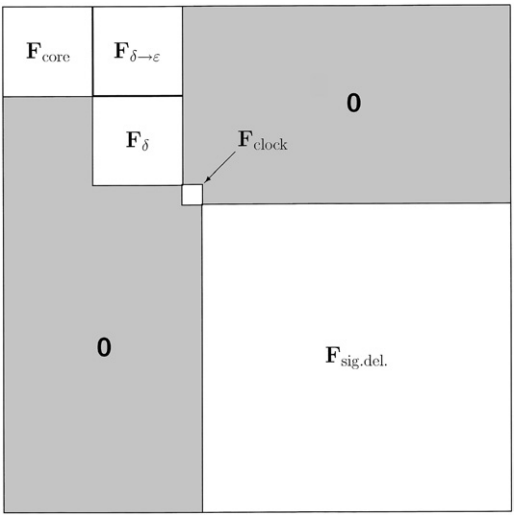


Fig. 12.2 Block structure of dynamic coefficient matrix.

updated in real time. Input axis misalignments, for example, tend to be relatively stable. If just the inertial sensor biases and scale factors (12 in all) will require updating, the total number would decrease to 9 + 12 + 2 + 12 = 35.

The model that is used in the rest of this chapter is for a gimbaled system, for which gyro scale factors are but a factor in the feedback gains used in stabilizing the platform and do not contribute significantly to navigation error. That would leave only 31 state variables. However, to simplify the state variable switching required as satellites come into use and fall out of use, this model includes the signal delays of all 31 satellites from the GPS constellation of July 7, 2012, rather than the 12-or-so used at any one time. That brings the total number of state variables to 9 + 9 + 2 + 31 = 51.

12.3.3.3 Dynamic Coefficient Matrix The resulting dynamic coefficient matrix for just those 51 state variables will have the structure shown in Fig. 12.2, where

- $\mathbf{F}_{\text{core}}$ is the 9×9 dynamic coefficient matrix for the core navigation errors—including vertical navigation errors.
- $\mathbf{F}_{\delta}$ is the 9×9 dynamic coefficient matrix for the nine sensor compensation parameters used. These include three each of accelerometer bias, accelerometer scale factor, and gyro bias.
- $\mathbf{F}_{\delta\to\varepsilon}$ is the dynamic coupling matrix of sensor compensation errors into navigation errors.

$\mathbf{F}_{\text{clock}}$ is the dynamic coefficient matrix for the receiver clock model. In this case, the model has only clock bias, but it can easily be expanded to include drift.

$\mathbf{F}_{\text{sig.del.}}$ is the dynamic coefficient matrix for satellite signal delay error due to unmodeled atmospheric effects.

Figure 12.2 shows the block structure of the dynamic coefficient matrix for the unified model, with sequential diagonal blocks representing the dynamics of navigation errors, sensor compensation errors, receiver clock errors, and atmospheric propagation delay errors, respectively. The contents of each of these blocks have already been derived.

12.3.3.4 Process Noise Covariance The unified model has process noise sources for

1. inertial sensor noise,
2. drifting inertial sensor compensation parameters,
3. receiver clock error, and
4. atmospheric propagation delays.

12.3.3.5 Measurement Sensitivities The sensitivity of pseudorange measurements to position is represented in terms of a unit vector in the direction from the satellite antenna to the receiver antenna. That is the essential "input axis" for the receiver as a pseudorange sensor. The longer the pseudorange, the more the estimated location should be adjusted in that direction. Each row of the measurement sensitivity matrix, corresponding to the pseudorange measurement from the ith satellite, would have the structure

$$\mathbf{H}_i = [u_{i\text{E}} \quad u_{i\text{N}} \quad u_{i\text{U}} \quad 0 \quad \cdots \quad 0 \quad 1 \quad 0 \quad \cdots \quad 0 \quad 1 \quad 0 \quad \cdots], \tag{12.14}$$

where

$$\mathbf{u}_i = \begin{bmatrix} u_{i\text{E}} \\ u_{i\text{N}} \\ u_{i\text{U}} \end{bmatrix} \tag{12.15}$$

is the ith pseudorange measurement input axis (a unit vector in the direction from the satellite to the antenna location); the first "1" after that is in the 19th column (clock bias), and the final "1" is in the $(20 + i)$th column (propagation delay for ith satellite).

Sensitivities of all pseudorange measurements to clock error are all equal to +1 if the clock error is in distance units and positive clock errors mean the clock is "fast."

12.4 PERFORMANCE ANALYSIS

12.4.1 Dynamic Simulation Model

12.4.1.1 State Transition Matrices (STMs) The dynamic coefficient matrix for navigation and sensor compensation errors has terms that depend on velocities and accelerations, so it is a time-varying system. In that case, the dynamic coefficient matrix must be evaluated at each simulation time step. If it were time-invariant, it would need to be computed only once. The lower 33×33 diagonal submatrix, representing clock and propagation delay errors, is time invariant. In this case, the block-diagonal structure of the dynamic coefficient can be exploited to make the computation involved a little easier.

For any block-diagonal matrix

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{11} & 0 & \cdots & 0 \\ 0 & \mathbf{F}_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{F}_{kk} \end{bmatrix}, \tag{12.16}$$

its equivalent state transition matrix for the discrete time step Δt is the matrix exponential

$$\mathbf{\Phi} = \exp(\Delta t \mathbf{F}) \tag{12.17}$$

$$= \begin{bmatrix} \exp(\Delta t \mathbf{F}_{11}) & 0 & \cdots & 0 \\ 0 & \exp(\Delta t \mathbf{F}_{22}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \exp(\Delta t \mathbf{F}_{kk}) \end{bmatrix}. \tag{12.18}$$

Consequently, if any of the diagonal blocks is time invariant, it only needs to be computed once. This is true for the dynamic coefficient submatrices for clock errors and signal propagation delay errors. The remaining time-varying block

$$\begin{bmatrix} \mathbf{F}_{\varepsilon}(v_{\mathrm{ENU}}(t)) & \mathbf{F}_{\delta \to \varepsilon}(a_{\mathrm{ENU}}(t)) \\ 0 & \mathbf{D}_{-1/\tau} \end{bmatrix}$$

has a lower-right time-invariant part, but the off-diagonal block $\mathbf{F}_{\delta \to \varepsilon}$ is time varying. Therefore, the upper-left 18×18 diagonal block of the dynamic coefficient matrix must be evaluated, multiplied by the time step, and transformed into its matrix exponential. Because the computational complexity of taking matrix exponentials increases as the cube of the dimension, this trick can save a lot of unnecessary computing.

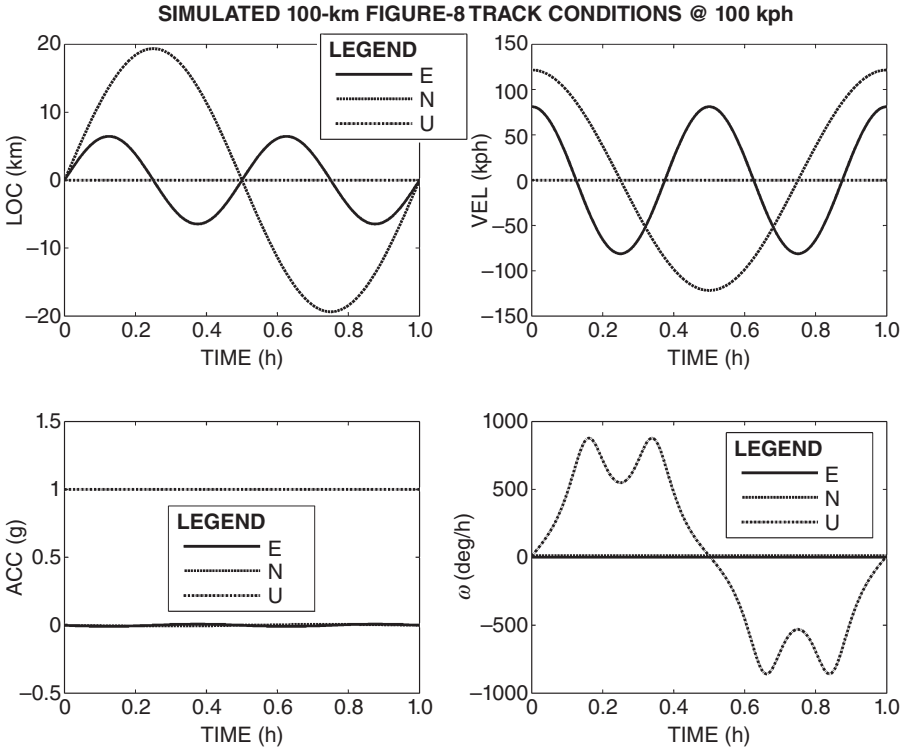


Fig. 12.3 Dynamic conditions on 100-km figure-8 test track.

12.4.1.2 Dynamic Simulation The MATLAB® function `Big8SimENU.m` on the accompanying website simulates dynamic conditions on a figure-eight track model similar to that used in Chapter 10, except that it has been scaled up by a factor of 60 to have a total track length of 100 km. The simulation has the same average speed of 100 kph, so the lap time is now 1 h. Dynamic test conditions are shown in Fig. 12.3, including position, velocity, acceleration, and inertial rotation rates. The rotation rates assume near-critical banking in the turns, as described in the simulation m-file `Big8SimENU`. However, these rates have little effect on gimbaled systems.

This same track simulation was used for navigation simulations using GNSS only, inertial navigation only, and integrated GNSS/INS.

12.4.2 Results

12.4.2.1 Stand-Alone GNSS Performance *Simulating Satellite Selection* GNSS receivers have their own built-in routines for selecting and using GNSS signals. Error simulation of GNSS navigation needs to simulate the receiver

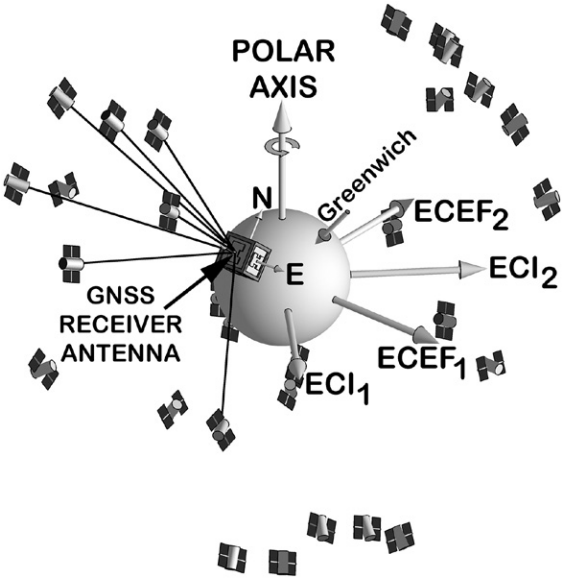


Fig. 12.4 GNSS navigation geometry.

methods. Doing so in locally level coordinates uses three different coordinate systems, as illustrated in Fig. 12.4:

1. Satellite ephemerides are in earth-centered inertial (ECI) coordinates.
2. The location of the GNSS receiver antenna is known from its own navigation solution, usually specified in earth-centered, earth-fixed (ECEF) coordinates in terms of latitude, longitude, and altitude.
3. The horizon mask at the receiver (for rejecting satellites too close to the horizon) is specified with respect to locally level coordinates. In Fig. 12.4, lines coming from the antenna (shown atop an INS, represented as a block in the figure) identify those satellites above a 15° horizon mask.

MATLAB® Implementation The MATLAB® m-file `GNSSonly.m` on the accompanying website runs a 6-h simulation using only GNSS with the DAMP3 vehicle tracking filter model described in Chapter 10. This filter includes estimates for the position, velocity, and acceleration of the receiver antenna. The filter model is statistically adjusted to track dynamics, using the track statistics determined by the m-file `Big8TrackStats.m`, the results of which are also listed as comments in `GNSSonly.m`. The empirically determined state dynamics are different for the east, north, and vertical components of position, velocity, and acceleration, and this information is used in tuning the filter to the application. The simulation includes a Kalman filter estimating the vehicle state. The program compares the simulated mean-squared position estimation

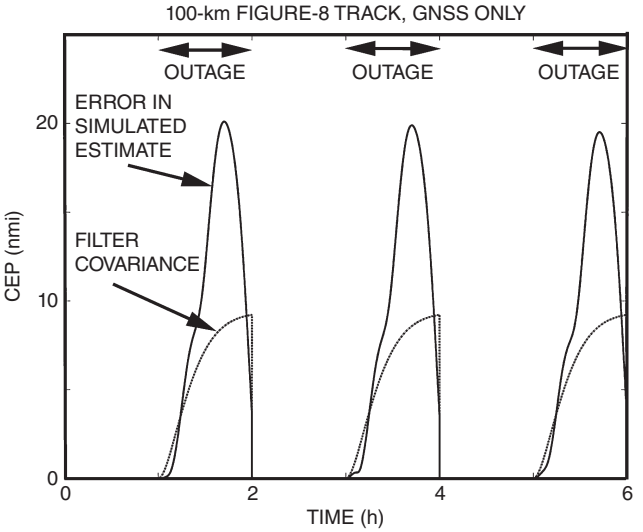


Fig. 12.5 GNSS-only navigation simulation results.

error to the values determined from the Riccati equations in the Kalman filter. The agreement is remarkably close when the GNSS receiver has satellite signal access but diverges significantly after signal outage.

The simulation was run with the satellite signals alternately available and unavailable, to show the tracking filter response to signal outage. Figure 12.5 is one of the plots generated by `GNSSonly.m`, showing the history of CEP based on the covariance of horizontal position error in the Kalman filter used for determining the navigation solution. Also shown is the estimate of CEP from simulating the actual navigation solution with simulated random noise sources.

12.4.2.2 Stand-Alone INS Performance The MATLAB® file `INS7only0.m` on the accompanying website simulates an INS under the same dynamic conditions as those in Fig. 12.5, with a set of sensor error parameters consistent with performance in the order of a nautical mile per hour. Figure 12.6 is one of its outputs. In order to avoid the problem of vertical channel instability, the 14-state error model includes only horizontal dynamics and sensor compensation errors for two horizontal accelerometers and three gyroscopes. In this case, GNSS signal outages have no effect on INS performance.

The MATLAB® file `INS7only0.m` also generates several plots showing the statistics of the different error variates over the 6-h simulated test run.

12.4.2.3 Integrated GNSS/INS Performance Figure 12.7 is a plot of the predicted performance of the same two systems from Fig. 12.5 and Fig. 12.6, but now integrated using a unified navigation model with 9 navigation

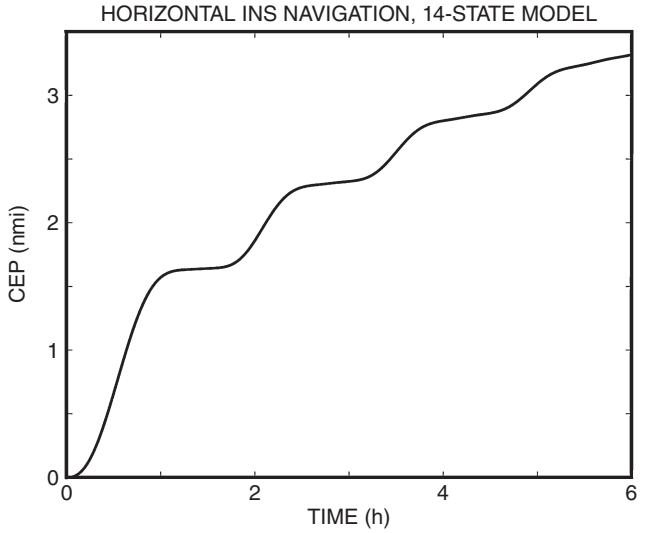


Fig. 12.6 INS-only navigation simulation results.

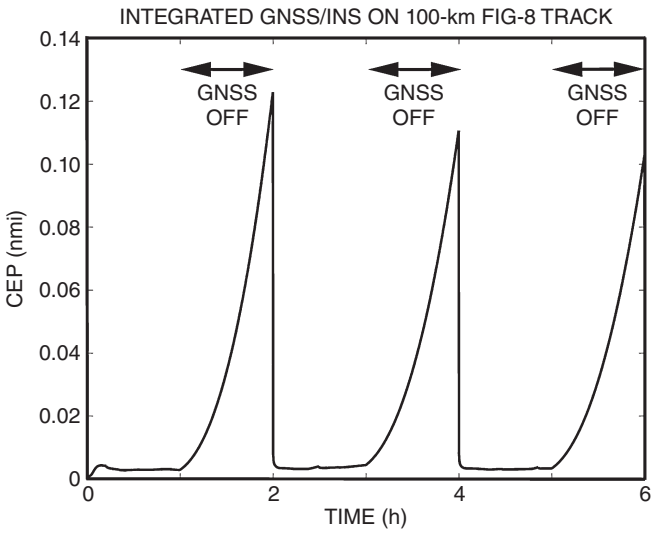


Fig. 12.7 Integrated GNSS/INS navigation simulation results.

variables, 9 sensor compensation variables, 2 GNSS receiver clock variables, and 31[4] GPS satellites.

Figures 12.5–12.7 capture the essential lesson here: Integrated GNSS/INS systems significantly outperform either of their independent subsystems.

Note that performance of the integrated system is superior to that of either subsystem when GNSS signals are available and even performs better than either when signals are unavailable. In this example, integrated performance during a 1-h signal outage is about 80 times better than from using GNSS alone. The secret to this is that the integrated system self-calibrates the inertial sensors when it has signal coverage, so that performance is much better when signals are lost.

The plot in Fig. 12.7—and many more—are generated by the MATLAB® m-file `GNSSINSInt0.m` on the accompanying website. The other plots show what is happening to the uncertainties of the various inertial sensor compensation variables when there is good GNSS signal coverage and how slowly things deteriorate when the signals are unavailable.

Those sensor performance parameters which determine integrated system performance are variables that can be altered in the m-file scripts to simulate a range of inertial navigators.

12.5 OTHER INTEGRATION ISSUES

12.5.1 Antenna/ISA Offset Correction

This is something that is insignificant in performance analysis but very important in the actual implementation.

When combining GNSS and INS navigation solutions, it is important to take into account the relative difference of location between the two. Each has its "holy point" at which the navigation solution is calculated, and this is different for the two navigation modes.

The holy point for a GNSS navigation receiver is its antenna. It is where the relative phases and transmission delays of all received signals are determined, and it is the location determined by the navigation solution. Any signal transmission and processing delays after the antenna contribute to latency of the solution, but not to errors in solution.

The INS holy point is somewhere within its inertial sensor assembly (ISA), which is where the accelerations and attitude rates of the host vehicle are measured and integrated.

The distance between the two navigation solutions can be large enough[5] that it must be taken into account when combining the two navigation

[4]GPS configuration of July 7, 2012, used in the simulations.

[5]It can be tens of meters for ships, where the INS may be located well below deck and the antenna is mounted high on a mast.

solutions. In that case, the displacement of the antenna from the ISA can be specified as a parameter vector,

$$\boldsymbol{\delta}_{\text{ant.,RPY}} = \begin{bmatrix} \delta_{\text{ant.,R}} \\ \delta_{\text{ant.,P}} \\ \delta_{\text{ant.,Y}} \end{bmatrix}, \tag{12.19}$$

in body-fixed roll–pitch–yaw (RPY) coordinates. Then, the displacement in east–north–up (ENU) coordinates will be

$$\boldsymbol{\delta}_{\text{ant.,ENU}} = \mathbf{C}_{\text{ENU}}^{\text{RPY}} \boldsymbol{\delta}_{\text{ant.,RPY}}, \tag{12.20}$$

where $\mathbf{C}_{\text{ENU}}^{\text{RPY}}$ is the coordinate transformation from RPY to ENU coordinates.

Usually, the matrix $\mathbf{C}_{\text{ENU}}^{\text{RPY}}$ and/or the roll, pitch, and yaw angles are variables in INS implementation software. Once $\boldsymbol{\delta}_{\text{ant.,ENU}}$ is computed, it can be used to relate the two navigation positions in ENU coordinates:

$$\mathbf{x}_{\text{ant.,ENU}} = \mathbf{x}_{\text{ISA,ENU}} + \boldsymbol{\delta}_{\text{ant.,ENU}} \tag{12.21}$$

$$\mathbf{x}_{\text{ISA,ENU}} = \mathbf{x}_{\text{ant.,ENU}} - \boldsymbol{\delta}_{\text{ant.,ENU}}, \tag{12.22}$$

eliminating the potential source of error.

This correction is generally included in integrated GNSS/INS systems. It requires a procedure for requesting and entering the components of $\boldsymbol{\delta}_{\text{ant.,RPY}}$ during system installation.

The antenna/ISA separation must also be taken into account in Doppler calculations for the antenna because rotation rates at the ISA can cause differential velocities and accelerations between the ISA and the receiver antenna.

12.5.2 Influence of Trajectories on Performance

Host vehicle dynamics influence integrated GNSS/INS navigation performance because some terms in the self-calibration model depend on attitudes, attitude rates, velocities, and accelerations. As a consequence, the observability of INS sensor compensation parameters from GNSS receiver pseudoranges depend on host vehicle dynamics. For example, the scale factors of sensors with zero input have no influence on the sensor outputs nor on navigation errors. This phenomenon was well understood half a century ago, when methods were developed for alignment of inertial systems aboard aircraft on carrier decks. The roll and pitch of the ship could be measured by the INS on the aircraft and compared to the measurements made in the ship's own INS.

However, this approach depends on their being some identifiable pitching and rolling. A similar situation arises in "transfer alignment" of inertial systems in stand-off weapons attached to aircraft wings, which may require selective maneuvering by the host aircraft to make this alignment observable.

The same is true of test conditions used for performance assessment of GNSS/INS systems. It is important that performance prediction be evaluated on representative trajectories from the intended missions. Systems intended for commercial aircraft should be evaluated on representative routes, and systems for military applications need to be evaluated under expected mission conditions. Each also needs to be evaluated for representative satellite geometries. Transpolar flights, for example, will have distinctively different satellite geometries.

The combined influence of host vehicle dynamics on integrated GNSS/INS performance can usually be characterized by a representative set of host vehicle trajectories, from which we can estimate the expected performance of the integrated GNSS/INS system over an ensemble of mission applications.

For example, a representative set of trajectories for a regional passenger jet might include several trajectories from gate to gate, including taxiing, takeoff, climb-out, cruise, midcourse heading and altitude changes, approach, landing, and taxiing. Trajectories with different total distances and headings should represent the expected range of applications. These can even be weighted according to the expected frequency of use. With such a set of trajectories, one can assess expected performances with different INS error characteristics and different satellite and pseudolite geometries.

Similarly, for a standoff air-to-ground weapon, an ensemble of trajectories with different approach and launch geometries and different target impact constraints can be used to evaluate RMS miss distances with GNSS jamming at different ranges from the target.

We demonstrated integrated GNSS/INS performance using a simple trajectory simulator, just to show the benefits of GNSS/INS integration. In order to quantify the expected performance for a specific application, however, you must use your own representative set of trajectories.

12.6 SUMMARY

1. GPS and INS were both developed for worldwide navigation capability, and together they have taken that capability to new levels of performance that neither approach could have achieved on its own.
2. The payoff in military costs and capabilities had driven development of GPS by the Department of Defense of the United States of America.
3. Some of its proponents and developers had foreseen many of the markets for GNSS/INS integration, including such applications as automating field equipment operations for farming and grading.

4. This insight came from using systems analysis of the integrated navigator to demonstrate its superior performance, just as we have done in this chapter.
5. The integrated system uses a unified navigation model for both types of sensors.
6. This model also includes the parameters used for compensating sensor output errors.
7. Sensor compensation parameters then become part of the navigation solution; that is, they are estimated right along with position and velocity, using the GNSS receiver pseudoranges and the inertial sensor outputs.
8. As a consequence, whenever GNSS signals are unavailable, improved accuracy of the sensor compensation parameters results in improved INS stand-alone navigation until GNSS signals become available again.
9. This has led to new automated control applications for integrated GNSS/INS systems, which require inertial sensors for precise and reliable operation under dynamic conditions. Integration with GNSS has brought the costs and capabilities of the resulting systems to very practical levels. The results of integrating inertial systems with GPS has made enormous improvements in achievable operational speed and efficiency.
10. GNSS systems architectures continue to change with the addition of more systems, more satellites, more signal channels, and more aiding systems, and integration-compatible inertial systems are also likely to continue to improve as the market expands, driving hardware costs further downward. It is a part of the Silicon Revolution, harnessing the enormous power and low cost of electronic systems to make our lives more enjoyable and efficient. As costs continue downward, the potential applications market continues to expand.
11. The MATLAB® m-files on the accompanying website can be easily modified by altering sensor performance parameters for exploring different INS design alternatives. They can also be altered to represent new GNSS constellations and signal characteristics, such as two-frequency methods for reducing atmospheric propagation delay errors.
12. This book is not intended to give you the answer to GNSS/INS integration. GNSS systems are changing constantly and every INS is unique in some way, so that answer is always changing. The purpose of this book is to show you how to find that answer yourself.

PROBLEM

12.1 Make a copy of the m-file `GNSSINSInt0.m` under another name so that it can be modified. Modify the renamed m-file to increase and decrease each of the following parameters by a factor of 10:

(a) RMS accelerometer noise

(b) RMS gyro noise

(c) RMS accelerometer bias

(d) RMS accelerometer scale factor

(e) RMS gyro bias.

Record the peak CEP during satellite signal outage for each case.

Questions:

(f) Which has the greatest influence on integrated GNSS/INS navigation performance?

(g) How linear is the relationship? That is, does RMS navigation performance go down by a factor of 10 when any of these is increased by a factor of 10?

(h) What does this say about which sensor performance specifications might be relaxed?

REFERENCES

[1] R. Babu and J. Wang, "Ultra-tight GPS/INS/PL Integration: Kalman Filter Performance Analysis," *GNSS*, Hong Kong, 2005, pp. 8–10.

[2] M. A. Lombardi, "The Accuracy and Stability of Quartz Watches," *Horological Journal* **150**(2), 57–59 (2008).

APPENDIX A

SOFTWARE

A.1 SOFTWARE SOURCES

The MATLAB® m-files on the accompanying website are intended to demonstrate to the reader how the methods described in this book actually work.

This is not "commercial grade" software, and it is not intended to be used as part of any commercial design process or product implementation software. The authors and publisher do not claim that this software meets any standards of merchantability, and we cannot assume any responsibility for the results if they are used for such purposes.

There is better, more reliable commercial software available for global navigation satellite system (GNSS) and inertial navigation system (INS) analysis, implementation, and integration. We have used the MATLAB® INS and Global Positioning System (GPS) toolboxes from GPSoft to generate some of the figures. There are also other commercial products available for these purposes. Many of the providers of such software maintain Internet websites describing their products and services, and the interested user is encouraged to search the Internet for suitable sources.

Most of the MATLAB® routines referred to in the text are self-documented by comments. The following sections contain short descriptions of those mentioned in the different chapters. Routines they call are also self-documented.

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

A.2 SOFTWARE FOR CHAPTER 3

`fBortz.m` computes the Bortz "noncommutative rate vector" as a function of measured body rates (ω) and cumulative rotation vector (ρ).

`Euler2CTMat.m` computes the coordinate transformation matrix represented by Euler angles.

`CTMat2Euler.m` computes the Euler angles equivalent to a given coordinate transformation matrix.

`RotVec2CTMat.m` computes the coordinate transformation matrix represented by a rotation vector.

`CTMat2RotVec.m` computes the rotation vector represented by a coordinate transformation matrix.

`RotVec2Quat.m` computes the rotation vector represented by a quaternion.

`Quat2RotVec.m` computes the quaternion represented by a rotation vector.

`Quat2CTMat.m` computes the coordinate transformation matrix represented by a quaternion.

`CTMat2Quat.m` computes the quaternion represented by a coordinate transformation matrix.

A.3 SOFTWARE FOR CHAPTER 4

The MATLAB® script `ephemeris.m` calculates a GPS satellite position in earth-centered, earth-fixed (ECEF) coordinates from its ephemeris parameters. The ephemeris parameters comprise a set of Keplerian orbital parameters and describe the satellite orbit during a particular time interval. From these parameters, ECEF coordinates are calculated using the equations from the text. Note that time t is the GPS time at transmission and t_k (tk) in the script is the total time difference between time t and the epoch time t_{oe} (toe). Kepler's equation for eccentric anomaly is nonlinear in E_k (Ek) and is solved numerically using the Newton–Raphson method.

The following MATLAB® scripts calculate satellite position for 24 h using almanac data on the accompanying website:

`GPS_position (PRN#)` plots satellite position using PRNs one at a time for all satellites.

`GPS_position_3D` plots satellite position for all PRNs in three dimensions. Use the rotate option in MATLAB® to see the satellite positions from the equator, the north pole, the south pole, and so on.

`GPS_el_az (PRN#, 33.8825,-117.8833)` plots satellite trajectory for a PRN from Fullerton, California (GPS laboratory located at California State University, Fullerton).

A.4 SOFTWARE FOR CHAPTER 7

The following MATLAB® scripts compute and plot ionospheric delays using Klobuchar models:

`Klobuchar_fix` plots the ionospheric delays for geostationary earth orbit (GEO) stationary satellites for 24 h, such as Anik F1R, GalaxyXV.

`Klobuchar (PRN#)` plots the ionospheric delays for a satellite specified by the argument PRN, when that satellite is visible.

`Iono_delay (PRN#)` plots the ionospheric delays for a PRN using dual-frequency data, when a satellite is visible. It uses the pseudorange carrier phase data for L1 and L2 signals. Plots are overlaid for comparison.

`init _var` initializes parameters and variables for `GPS_perf.m`.

`GPS_perf.m` performs covariance analysis of the expected performance of a GPS receiver using a Kalman filter.

`calcH` calculates the **H** matrix for `GPS_perf.m`.

`gdop` calculates the geometric dilution of precision (GDOP) for chosen constellation for `GPS_perf.m`.

`covar` solves the Riccati equation for `GPS_perf.m`.

`plot_covar` plots results from `GPS_perf.m`.

A.5 SOFTWARE FOR CHAPTER 10

`osc_ekf.m` demonstrates an extended Kalman filter tracking the phase, amplitude, frequency, and damping factor of a harmonic oscillator with randomly time-varying parameters.

`SchmidtKalmanTest.m` compares Schmidt–Kalman filter and Kalman filter for GPS navigation with time-correlated pseudorange errors.

`shootout.m` compares performance of several square root covariance filtering methods on an ill-conditioned problem from P. Dyer and S. McReynolds, "Extension of Square-Root Filtering to Include Process Noise," *Journal of Optimization Theory and Applications* **3**, 444–458 (1969).

`Joseph`, called by `shootout.m`, implements the "Joseph stabilized" Kalman filter.

`Josephb`, called by `shootout.m`, implements the "Joseph–Bierman" Kalman filter.

`Josephdv`, called by `shootout.m`, implements the "Joseph–DeVries" Kalman filter.

`Potter`, called by `shootout.m`, implements the Potter square-root filter.

`Carlson`, called by `shootout.m`, implements the Carlson square-root filter.

`Bierman`, called by `shootout.m`, implements the Bierman square-root filter.

`Damp2eval.m` evaluates DAMP2 GPS position tracking filters for a range of RMS accelerations and acceleration correlation times.

`Damp2Params.m` generates parameters for the DAMP2 vehicle tracking model, based on statistics of the vehicle trajectories.

`DAMP3Params.m` generates parameters for the DAMP3 vehicle tracking model, based on statistics of the vehicle trajectories.

GPSTrackingDemo.m applies the GPS vehicle tracking filters TYPE2, DAMP2, DAMP3, and FIG8 to the same problem (tracking a vehicle moving on a figure-eight test track).

Fig 8TrackDemo.m generates a series of plots and statistics of the simulated figure-eight test track trajectory.

Fig 8Mod1D.m simulates the trajectory of a vehicle going around a figure-eight test track. GPS_exampl.m implements Problem 10.7.

A.6 SOFTWARE FOR CHAPTER 11

Fcore9.m calculates the 9×9 dynamic coefficient matrix for the nine "core" navigation error variables.

Fcore10Test1.m simulates barometric altimeter damping of INS, using the Fcore9 model for INS errors, plus a model for altimeter errors due to ambient barometric pressure variations.

DampingTest10part1.m performs covariance analysis of altimeter damping vertical channels of INS, using a nine-state model of INS errors, plus an altimeter with exponentially correlated bias.

Fcore7.m calculates the 7×7 dynamic coefficient matrix for horizontal navigation error variables.

Fcore7Test1.m through Fcore7Test7.m use Fcore7.m to perform a series of tests to verify the model.

CEPvsSensorNoise.m performs a series of dynamic simulations to characterize INS CEP rate as a function of accelerometer and gyroscope noise.

CEPvsAccComp.m performs a series of dynamic simulations to characterize INS CEP rate as a function of accelerometer bias and scale factor compensation errors.

A.7 SOFTWARE FOR CHAPTER 12

Damp2Params.m solves a transcendental equation for alternative parameters in the DAMP2 GPS tracking filter.

Damp3Params.m solves a transcendental equation for alternative parameters in the DAMP3 GPS tracking filter.

GPSTrackingDemo.m applies the GPS vehicle tracking filters TYPE2, DAMP2, DAMP3, and FIG8 to the same problem (tracking a vehicle moving on a figure-eight test track).

Fig 8TrackDemo.m generates a series of plots and statistics of the simulated figure-eight test track trajectory.

HorizINSperfModel.m calculates INS error model parameters as a function of CEP rate.

GPSINSwGPSpos.m simulates GPS/INS loosely coupled integration using only standard GPS and INS output position values.

`GPSI NSwPRs.m` simulates GPS/INS tightly coupled integration using GPS pseudoranges and INS position outputs.

A.8 ALMANAC/EPHEMERIS DATA SOURCES

All ephemerides are for GPS satellites, which have been the only sources at times. Several ephemeris sources are used for generating satellite positions as a function of time. Some are m-files with right ascensions, arguments of perigee, and satellite angles stored in matrices. These were downloaded from government sources listed in the comments. For example,

`YUMAdata.m` loads GPS almanac data from the U.S. Coast Guard website for GPS almanacs for Wednesday, March 08, 2006 10:48 a.m., converts to arrays of right ascension and phase angles for 29 satellites (used by `Damp2eval.m`).

In addition, more recent downloads from the U.S. Coast Guard website have been downloaded in ASCII, converted to MATLAB® arrays of right ascension and the sum of argument of perigee and satellite angle, and converted to "`.dat`" files using the MATLAB® save command. Two of these are named `RA.dat` (right ascensions) and `PA.dat` (perigee angle plus satellite angle). The sum of perigee angle and satellite angle represents the satellite location with respect to the south-to-north equatorial plane crossing at the time the prime meridian is at the vernal equinox.

The MATLAB® script `FetchYUMAdata.m` converts ASCII files downloaded from the U.S. Coast Guard website, which you can use to obtain more recent ephemerides. The ASCII file `YUMAdata.txt` was downloaded from this website on July 7, 2012 and was used for creating the files `RA.dat` and `PA.dat` on the accompanying website. See the file `YUMAdata.txt` for an example of the data downloaded. Instructions for navigating the Coast Guard website http://www.navcen.uscg.gov/?pageName=gpsAlmanacs to download the current GPS almanac are given in the comments at the end of `YUMAdata.txt`.

APPENDIX B

COORDINATE SYSTEMS AND TRANSFORMATIONS

Navigation makes use of coordinates that are natural to the problem at hand: inertial coordinates for inertial navigation, orbital coordinates for global navigation satellite system (GNSS) navigation, and earth-fixed coordinates for representing locations on the earth.

The principal coordinate systems used, and the transformations between these different coordinate systems, are summarized in this appendix. These are primarily Cartesian (orthogonal) coordinates, and the transformations between them can be represented by orthogonal matrices. However, the coordinate transformations can also be represented by rotation vectors or quaternions, and all representations are used in the derivations and implementation of GNSS/inertial navigation system (INS) integration.

B.1 COORDINATE TRANSFORMATION MATRICES

B.1.1 Notation

We use the notation $\mathbf{C}_{\text{TO}}^{\text{FROM}}$ to denote a coordinate transformation matrix from one coordinate frame (designated by "FROM") to another coordinated frame (designated by "TO"). For example,

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

$\mathbf{C}_{\text{ENU}}^{\text{ECI}}$ denotes the coordinate transformation matrix from earth-centered inertial (ECI) coordinates (Section B.2.2) to earth-fixed, locally level, east–north–up (ENU) coordinates (Section B.7.2).

$\mathbf{C}_{\text{NED}}^{\text{RPY}}$ denotes the coordinate transformation matrix from vehicle body-fixed roll–pitch–yaw (RPY) coordinates (Section B.3.8) to earth-fixed north–east–down (NED) coordinates (Section B.3.7.2).

B.1.2 Definitions

What we mean by a coordinate transformation matrix is that if a vector $\mathbf{v}$ has the representation

$$\mathbf{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{B.1}$$

in XYZ coordinates and the same vector $\mathbf{v}$ has the alternative representation

$$\mathbf{v} = \begin{bmatrix} v_u \\ v_v \\ v_w \end{bmatrix} \tag{B.2}$$

in UVW coordinates, then

$$\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \mathbf{C}_{\text{XYZ}}^{\text{UVW}} \begin{bmatrix} v_u \\ v_v \\ v_w \end{bmatrix}, \tag{B.3}$$

where XYZ and UVW stand for any two Cartesian coordinate systems in three-dimensional space.

B.1.3 Unit Coordinate Vectors

The components of a vector in either coordinate system can be expressed in terms of the vector components along unit vectors parallel to the respective coordinate axes. For example, if one set of coordinate axes is labeled X, Y, and Z, and the other set of coordinate axes is labeled U, V, and W, then the same vector $\mathbf{v}$ can be expressed in either coordinate frame as

$$\mathbf{v} = v_x \vec{\mathbf{1}}_x + v_y \vec{\mathbf{1}}_y + v_z \vec{\mathbf{1}}_z \tag{B.4}$$

$$= v_u \vec{\mathbf{1}}_u + v_v \vec{\mathbf{1}}_v + v_w \vec{\mathbf{1}}_w, \tag{B.5}$$

where

- the unit vectors $\vec{\mathbf{1}}_x$, $\vec{\mathbf{1}}_y$, and $\vec{\mathbf{1}}_z$ are along the XYZ axes;
- the scalars v_x, v_y, and v_z are the respective components of $\mathbf{v}$ along the XYZ axes;
- the unit vectors $\vec{\mathbf{1}}_u$, $\vec{\mathbf{1}}_v$, and $\vec{\mathbf{1}}_w$ are along the UVW axes; and
- the scalars v_u, v_v, and v_w are the respective components of $\mathbf{v}$ along the UVW axes.

B.1.4 Direction Cosines

The respective components can also be represented in terms of dot products of $\mathbf{v}$ with the various unit vectors,

$$v_x = \vec{\mathbf{1}}_x^T \mathbf{v} = v_u \vec{\mathbf{1}}_x^T \vec{\mathbf{1}}_u + v_v \vec{\mathbf{1}}_x^T \vec{\mathbf{1}}_v + v_w \vec{\mathbf{1}}_x^T \vec{\mathbf{1}}_w \tag{B.6}$$

$$v_y = \vec{\mathbf{1}}_y^T \mathbf{v} = v_u \vec{\mathbf{1}}_y^T \vec{\mathbf{1}}_u + v_v \vec{\mathbf{1}}_y^T \vec{\mathbf{1}}_v + v_w \vec{\mathbf{1}}_y^T \vec{\mathbf{1}}_w \tag{B.7}$$

$$v_z = \vec{\mathbf{1}}_z^T \mathbf{v} = v_u \vec{\mathbf{1}}_z^T \vec{\mathbf{1}}_u + v_v \vec{\mathbf{1}}_z^T \vec{\mathbf{1}}_v + v_w \vec{\mathbf{1}}_z^T \vec{\mathbf{1}}_w, \tag{B.8}$$

which can be represented in matrix form as

$$\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} \vec{\mathbf{1}}_x^T \vec{\mathbf{1}}_u & \vec{\mathbf{1}}_x^T \vec{\mathbf{1}}_v & \vec{\mathbf{1}}_x^T \vec{\mathbf{1}}_w \\ \vec{\mathbf{1}}_y^T \vec{\mathbf{1}}_u & \vec{\mathbf{1}}_y^T \vec{\mathbf{1}}_v & \vec{\mathbf{1}}_y^T \vec{\mathbf{1}}_w \\ \vec{\mathbf{1}}_z^T \vec{\mathbf{1}}_u & \vec{\mathbf{1}}_z^T \vec{\mathbf{1}}_v & \vec{\mathbf{1}}_z^T \vec{\mathbf{1}}_w \end{bmatrix} \begin{bmatrix} v_u \\ v_v \\ v_w \end{bmatrix} \tag{B.9}$$

$$\stackrel{\text{def}}{=} \mathbf{C}_{\text{XYZ}}^{\text{UVW}} \begin{bmatrix} v_u \\ v_v \\ v_w \end{bmatrix}, \tag{B.10}$$

which defines the coordinate transformation matrix $\mathbf{C}_{\text{XYZ}}^{\text{UVW}}$ from UVW to XYZ coordinates in terms of the dot products of unit vectors. However, dot products of unit vectors also satisfy the cosine rule:

$$v^{\mathrm{T}} w = |v||w| \cos(\theta_{ab}), \tag{B.11}$$

where θ_{ab} is the angle between the vectors v and w. For unit vectors, this has the form

$$\vec{\mathbf{1}}_a^{\mathrm{T}} \vec{\mathbf{1}}_b = \cos(\theta_{ab}), \tag{B.12}$$

where θ_{ab} is the angle between $\vec{\mathbf{1}}_a$ and $\vec{\mathbf{1}}_b$. As a consequence, the coordinate transformation matrix can also be written in the form

$$\mathbf{C}_{\mathrm{XYZ}}^{\mathrm{UVW}} = \begin{bmatrix} \cos(\theta_{xu}) & \cos(\theta_{xv}) & \cos(\theta_{xw}) \\ \cos(\theta_{yu}) & \cos(\theta_{yv}) & \cos(\theta_{yw}) \\ \cos(\theta_{zu}) & \cos(\theta_{zv}) & \cos(\theta_{zw}) \end{bmatrix}, \tag{B.13}$$

which is why coordinate transformation matrices are also called **direction cosine matrices**.

B.1.5 Composition of Coordinate Transformations

Coordinate transformation matrices satisfy the composition rule

$$\mathbf{C}_{\mathrm{C}}^{\mathrm{B}}\mathbf{C}_{\mathrm{B}}^{\mathrm{A}} = \mathbf{C}_{\mathrm{C}}^{\mathrm{A}},$$

where A, B, and C represent different coordinate frames.

B.2 INERTIAL REFERENCE DIRECTIONS

Celestial reference directions were established when the earth was thought to be the center of the universe, with the sun, stars, and planets revolving around it. The apparent rotation axis of the stars, the North Pole was adopted as a reference direction. A second reference was defined by the apparent direction from Earth to its sun when the sun appeared to be crossing the equatorial plane (the plane orthogonal to the earth's polar axis), entering the northern hemisphere. This happens at the time of year we call the **vernal equinox**, which usually occurs around March 21–23.

These are less-than-perfect choices for inertial directions, however. The inertial direction of the rotation axis of Earth precesses around a cone with angular radius of about 23.5°, and with a period of about 25,772 years. The inertial rotation rate of this supposedly fixed reference direction is around 1.6×10^{-6} deg/h. The primary cause is the torque on the earth due to the gravity gradients from the sun and moon acting on the earth's equatorial bulge. There are other, less predictable variations in the inertial direction of the earth's spin axis from such sources as the angular momentum in storms and ocean currents, or changes in the density distribution of Earth due to tectonic events and global warming. These effects have timescales ranging from days to millenia.

Variations in the defined inertial directions are usually corrected by giving a particular date and time for the assumed value of the mean equatorial plane.

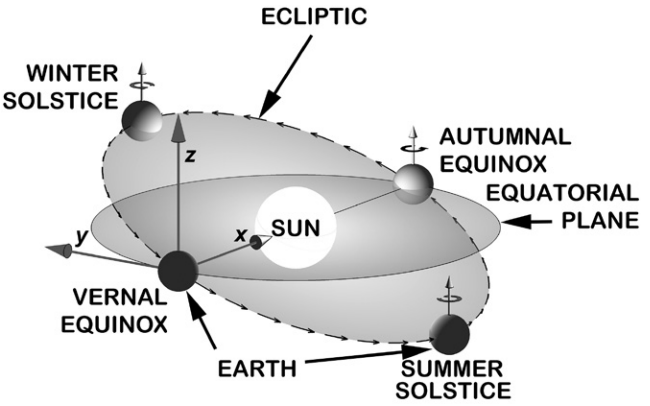


Fig. B.1 Earth-centered inertial (ECI) coordinates.

A third orthogonal coordinate axis can be defined by the cross product of the first two, such that the three axes define a right-handed orthogonal coordinate system. A heliocentric view of these directions is illustrated in Fig. B.1.

The orbital plane of Earth around the sun (once considered heretical) is called the **ecliptic**. The vernal equinox direction is parallel to the intersection of the ecliptic and the equatorial plane of the earth.

B.3 APPLICATION-DEPENDENT COORDINATE SYSTEMS

Although we are concerned exclusively with coordinate systems in the three dimensions of the observable world, there are many ways of representing a location in that world by a set of coordinates. The coordinates presented here are those used in navigation with GNSS and/or INS.

B.3.1 Cartesian and Polar Coordinates

René Descartes (1596–1650) introduced the idea of representing points in three-dimensional space by a triplet of coordinates, called "Cartesian coordinates," in his honor. They are also called "Euclidean coordinates." The Cartesian coordinates (x, y, z) and polar coordinates (θ, ϕ, r) of a common reference point, as illustrated in Fig. B.2, are related by the equations

$$x = r\cos(\theta)\cos(\phi) \tag{B.14}$$

$$y = r\sin(\theta)\cos(\phi) \tag{B.15}$$

$$z = r\sin(\phi) \tag{B.16}$$

$$r = \sqrt{x^2 + y^2 + z^2} \tag{B.17}$$

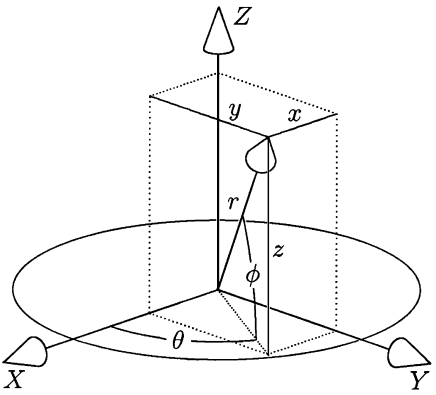


Fig. B.2 Cartesian and polar coordinates.

$$\phi = \arcsin\left(\frac{z}{r}\right) (-\pi/2 \le \phi \le +\pi/2) \tag{B.18}$$

$$\theta = \arctan\left(\frac{y}{x}\right) (-\pi < \theta \le +\pi), \tag{B.19}$$

with the angle θ (in radians) undefined if $\phi = \pm\pi/2$.

B.3.2 Celestial Coordinates

Similar to ECI coordinates, the "celestial sphere" is a quasi-inertial polar coordinate system referenced to the polar axis of the earth and the vernal equinox. The prime meridian of the celestial sphere is fixed to the vernal equinox. Polar celestial coordinates are **right ascension** (RA) (the celestial analog of longitude, measured eastward from the vernal equinox) and **declination** (the celestial analog of latitude), as illustrated in Fig. B.3. Because the celestial sphere is used primarily as a reference for direction, no origin needs to be specified.

The RA is zero at the vernal equinox and increases eastward (in the direction the earth turns). The units of RA can be radian, degree, or hour (with 15 deg/h as the conversion factor).

By convention, declination is zero in the equatorial plane and increases toward the North Pole, with the result that celestial objects in the northern hemisphere have positive declinations. Its units can be degree or radian.

B.3.3 Satellite Orbit Coordinates

Johannes Kepler (1571–1630) discovered the geometrical shapes of the orbits of planets, and the minimum number of parameters necessary to specify an orbit (called "Keplerian" parameters). Keplerian parameters used to specify

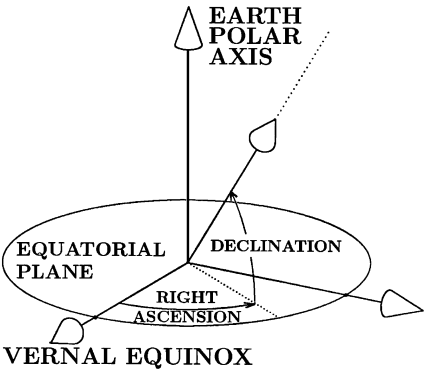


Fig. B.3 Celestial coordinates.

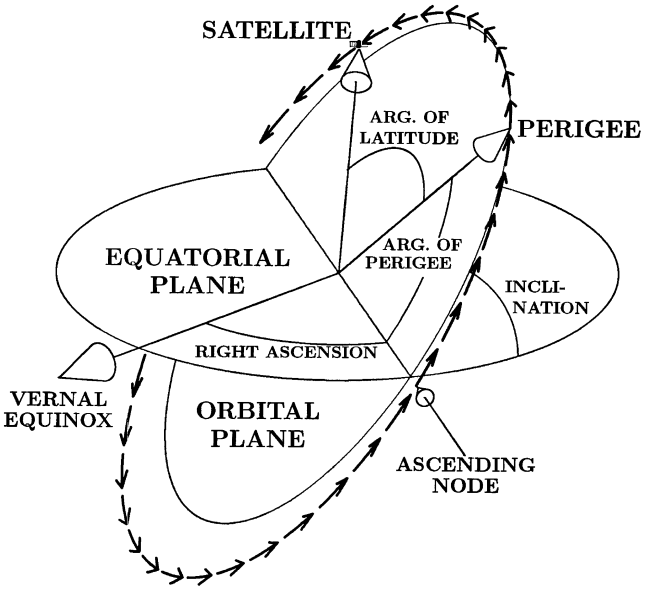


Fig. B.4 Keplerian parameters for satellite orbit.

GNSS satellite orbits in terms of their orientations relative to the equatorial plane and the vernal equinox (defined Section B.2.1 and illustrated in Fig. B.1) include

- The RA of the ascending node and orbit inclination, specifying the orientation of the orbital plane with respect to the vernal equinox and equatorial plane, as illustrated in Fig. B.4:
  - RA is defined in Section B.3.2 and shown in Fig. B.3.

  - The intersection of the orbital plane of a satellite with the equatorial plane is called its "line of nodes," where the "nodes" are the two intersections of the satellite orbit with this line. The two nodes are dubbed "ascending[1]" (i.e., ascending from the southern hemisphere to the northern hemisphere) and "descending." The right ascension of the ascending node (RAAN) is the angle in the equatorial plane from the vernal equinox to the ascending node, measured counterclockwise as seen looking down from the North Pole direction.
  - Orbital inclination is the dihedral angle between the orbital plane and the equatorial plane. It ranges from 0° (orbit in equatorial plane) to 90° (polar orbit).
- Semimajor axis a and semiminor axis b (defined in Section B.3.5.4 and illustrated in Fig. B.6), specifying the size and shape of the elliptical orbit within the orbital plane
- Orientation of the ellipse within its orbital plane, specified in terms of the "argument of perigee," the angle between the ascending node and the perigee of the orbit (closest approach to Earth), as illustrated in Fig. B.4
- Position of the satellite relative to perigee of the elliptical orbit, specified in terms of the angle from perigee—called the "argument of latitude" or "true anomaly," both illustrated in Fig. B.4.

For computer simulation demonstrations, GNSS satellite orbits can usually be assumed to be circular with radius $a = b = R = 26{,}560$ km and inclined at 55° to the equatorial plane. This eliminates the need to specify the orientation of the elliptical orbit within the orbital plane. (The argument of perigee becomes overly sensitive to orbit perturbations when eccentricity is close to zero.)

B.3.4 ECI Coordinates

The ECI coordinates illustrated in Fig. B.1 are the favored inertial coordinates in the near-earth environment. The origin of ECI coordinates is at the center of gravity of the earth, with axis directions

x, in the direction of the vernal equinox;
z, parallel to the rotation axis (north polar axis) of the earth; and
y, an additional axis to make this a right-handed orthogonal coordinate system, as illustrated in Fig. B.1.

The equatorial plane of the earth is also the equatorial plane of ECI coordinates, but the earth itself is rotating relative to the vernal equinox at its sidereal

[1]The astronomical symbol for the ascending node is ☊, often read as "earphones."

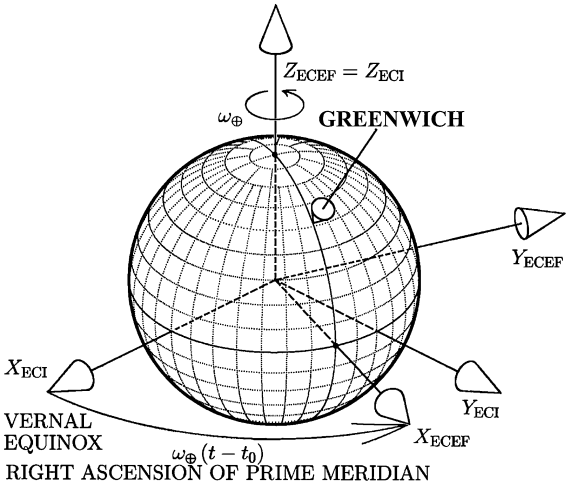


Fig. B.5 ECI and ECEF coordinates.

rotation rate of about 7,292,115,167 × 10^{-14} rad/s, or about 15.04109 deg/h, as illustrated in Fig. B.5.

B.3.5 Earth-Centered, Earth-Fixed (ECEF) Coordinates

ECEF coordinates have the same origin (earth center) and third (polar) axis as ECI coordinates, but rotate with the earth—as shown in Fig. B.5. As a consequence, ECI and ECEF longitudes differ only by a function of time.

B.3.5.1 Longitudes in ECEF Coordinates Longitudes in ECEF coordinates are measured east (+) and west (–) from the prime meridian passing through the principal transit instrument at the observatory at Greenwich, United Kingdom, a convention adopted by 41 representatives of 25 nations at the International Meridian Conference, held in Washington, DC, in October of 1884.

B.3.5.2 Latitudes in ECEF Coordinates **Latitudes** are measured with respect to the equatorial plane, but there is more than one kind of "latitude" on the planet.

Geocentric latitude would be measured as the angle between the equatorial plane and a line from the reference point to the center of the earth, but this angle could not be determined accurately (before GNSS) without running a transit survey over vast distances.

The angle between the pole star and the local gravitational vertical direction can be measured more readily, and that angle is more closely approxi-

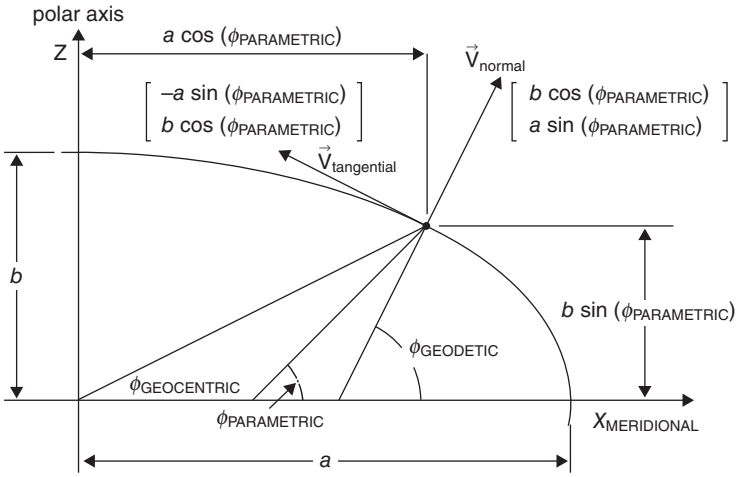


Fig. B.6 Geocentric, parametric, and geodetic latitudes in meridional plane.

mated as *geodetic latitude*. There is yet a third latitude (parametric latitude) that is useful in analysis. These alternative latitudes are defined in the following subsections.

B.3.5.3 Latitude on an Ellipsoidal Earth *Geodesy* is the study of the size and shape of the earth, and the establishment of physical control points defining the origin and orientation of coordinate systems for mapping the earth. Earth-shape models are very important for navigation using either GNSS or INS, or both. INS alignment is with respect to the local vertical, which does not generally pass through the center of the earth. That is because the earth is not spherical and it is rotating.

At different times in history, the earth has been regarded as being flat (first-order approximation), spherical (second-order), and ellipsoidal (third-order). The third-order model is an ellipsoid of revolution, with its shorter radius at the poles and its longer radius at the equator.

B.3.5.4 Parametric Latitude For geoids based on ellipsoids of revolution, every meridian is an ellipse with equatorial radius a (also called "semimajor axis") and polar radius b (also called "semiminor axis"). If we let z be the Cartesian coordinate in the polar direction and $x_{\text{MERIDIONAL}}$ be the equatorial coordinate in the meridional plane, as illustrated in Fig. B.6, then the equation for this ellipse will be

$$\frac{x^2_{\text{MERIDIONAL}}}{a^2} + \frac{z^2}{b^2} = 1 \tag{B.20}$$

$$1 = \cos^2(\phi_{\text{PARAMETRIC}}) + \sin^2(\phi_{\text{PARAMETRIC}}) \tag{B.21}$$

$$= \frac{a^2 \cos^2(\phi_{\text{PARAMETRIC}})}{a^2} + \frac{b^2 \sin^2(\phi_{\text{PARAMETRIC}})}{b^2} \tag{B.22}$$

$$= \frac{\{a\cos(\phi_{\text{PARAMETRIC}})\}^2}{a^2} + \frac{\{b\sin(\phi_{\text{PARAMETRIC}})\}^2}{b^2}; \tag{B.23}$$

that is, a parametric solution for the ellipse is

$$x_{\text{MERIDIONAL}} = a\cos(\phi_{\text{PARAMETRIC}}) \tag{B.24}$$

$$z = b\sin(\phi_{\text{PARAMETRIC}}), \tag{B.25}$$

as illustrated in Fig. B.6. Although the parametric latitude $\phi_{\text{PARAMETRIC}}$ has no physical significance, it is quite useful for relating geocentric and geodetic latitude, which do have physical significance.

B.3.5.5 Geodetic Latitude Geodetic latitude is defined as the elevation angle above (+) or below (–) the equatorial plane of the normal to the ellipsoidal surface. This direction can be defined in terms of the parametric latitude because it is orthogonal to the meridional tangential direction.

The vector tangential to the meridian will be in the direction of the derivative to the elliptical equation solution with respect to parametric latitude:

$$\vec{\mathbf{v}}_{\text{tangential}} \propto \frac{\partial}{\partial\phi_{\text{PARAMETRIC}}}\begin{bmatrix} a\cos(\phi_{\text{PARAMETRIC}}) \\ b\sin(\phi_{\text{PARAMETRIC}}) \end{bmatrix} \tag{B.26}$$

$$= \begin{bmatrix} -a\sin(\phi_{\text{PARAMETRIC}}) \\ b\cos(\phi_{\text{PARAMETRIC}}) \end{bmatrix}, \tag{B.27}$$

and the meridional normal direction will be orthogonal to it, or

$$\vec{\mathbf{v}}_{\text{normal}} \propto \begin{bmatrix} b\cos(\phi_{\text{PARAMETRIC}}) \\ a\sin(\phi_{\text{PARAMETRIC}}) \end{bmatrix}, \tag{B.28}$$

as illustrated in Fig. B.5.

The tangent of geodetic latitude is then the ratio of the z and x components of the surface normal vector, or

$$\tan(\phi_{\text{GEODETIC}}) = \frac{a\sin(\phi_{\text{PARAMETRIC}})}{b\cos(\phi_{\text{PARAMETRIC}})} \tag{B.29}$$

$$= \frac{a}{b}\tan(\phi_{\text{PARAMETRIC}}), \tag{B.30}$$

from which, using some standard trigonometric identities,

$$\sin(\phi_{\text{GEODETIC}}) = \frac{\tan(\phi_{\text{GEODETIC}})}{\sqrt{1+\tan^2(\phi_{\text{GEODETIC}})}} \tag{B.31}$$

$$= \frac{a\sin(\phi_{\text{PARAMETRIC}})}{\sqrt{a^2\sin^2(\phi_{\text{PARAMETRIC}}) + b^2\cos^2(\phi_{\text{PARAMETRIC}})}} \tag{B.32}$$

$$\cos(\phi_{\text{GEODETIC}}) = \frac{1}{\sqrt{1+\tan^2(\phi_{\text{GEODETIC}})}} \tag{B.33}$$

$$= \frac{b\cos(\phi_{\text{PARAMETRIC}})}{\sqrt{a^2\sin^2(\phi_{\text{PARAMETRIC}}) + b^2\cos^2(\phi_{\text{PARAMETRIC}})}}. \tag{B.34}$$

The inverse relationship is

$$\tan(\phi_{\text{PARAMETRIC}}) = \frac{b}{a}\tan(\phi_{\text{GEODETIC}}), \tag{B.35}$$

from which, using the same trigonometric identities as before,

$$\sin(\phi_{\text{PARAMETRIC}}) = \frac{\tan(\phi_{\text{PARAMETRIC}})}{\sqrt{1+\tan^2(\phi_{\text{PARAMETRIC}})}} \tag{B.36}$$

$$= \frac{b\sin(\phi_{\text{GEODETIC}})}{\sqrt{a^2\cos^2(\phi_{\text{GEODETIC}}) + b^2\sin^2(\phi_{\text{GEODETIC}})}} \tag{B.37}$$

$$\cos(\phi_{\text{PARAMETRIC}}) = \frac{1}{\sqrt{1+\tan^2(\phi_{\text{PARAMETRIC}})}} \tag{B.38}$$

$$= \frac{a\cos(\phi_{\text{GEODETIC}})}{\sqrt{a^2\cos^2(\phi_{\text{GEODETIC}}) + b^2\sin^2(\phi_{\text{GEODETIC}})}}, \tag{B.39}$$

and the 2D X-Z Cartesian coordinates in the meridional plane of a point on the geoid surface will be

$$x_{\text{MERIDIONAL}} = a\cos(\phi_{\text{PARAMETRIC}}) \tag{B.40}$$

$$= \frac{a^2\cos(\phi_{\text{GEODETIC}})}{\sqrt{a^2\cos^2(\phi_{\text{GEODETIC}}) + b^2\sin^2(\phi_{\text{GEODETIC}})}} \tag{B.41}$$

$$z = b\sin(\phi_{\text{PARAMETRIC}}) \tag{B.42}$$

$$= \frac{b^2\sin(\phi_{\text{GEODETIC}})}{\sqrt{a^2\cos^2(\phi_{\text{GEODETIC}}) + b^2\sin^2(\phi_{\text{GEODETIC}})}} \tag{B.43}$$

in terms of geodetic latitude.

Equations B.41 and B.43 apply only to points on the geoid surface. Orthometric height h above (+) or below (–) the geoid surface is measured along

the surface normal, so that the X-Z coordinates for a point with altitude h will be

$$x_{\text{MERIDIONAL}} = \cos(\phi_{\text{GEODETIC}}) \times \left\{ h + \frac{a^2}{\sqrt{a^2 \cos^2(\phi_{\text{GEODETIC}}) + b^2 \sin^2(\phi_{\text{GEODETIC}})}} \right\} \tag{B.44}$$

$$z = \sin(\phi_{\text{GEODETIC}}) \times \left\{ h + \frac{b^2}{\sqrt{a^2 \cos^2(\phi_{\text{GEODETIC}}) + b^2 \sin^2(\phi_{\text{GEODETIC}})}} \right\}. \tag{B.45}$$

In 3D ECEF coordinates, with X-axis passing through the equator at the prime meridian (at which longitude $\theta = 0$),

$$x_{\text{ECEF}} = \cos(\theta) x_{\text{MERIDIONAL}} \tag{B.46}$$

$$= \cos(\theta) \cos(\phi_{\text{GEODETIC}}) \times \left\{ h + \frac{a^2}{\sqrt{a^2 \cos^2(\phi_{\text{GEODETIC}}) + b^2 \sin^2(\phi_{\text{GEODETIC}})}} \right\} \tag{B.47}$$

$$y_{\text{ECEF}} = \sin(\theta) x_{\text{MERIDIONAL}} \tag{B.48}$$

$$= \sin(\theta) \cos(\phi_{\text{GEODETIC}}) \times \left\{ h + \frac{a^2}{\sqrt{a^2 \cos^2(\phi_{\text{GEODETIC}}) + b^2 \sin^2(\phi_{\text{GEODETIC}})}} \right\} \tag{B.49}$$

$$z_{\text{ECEF}} = \sin(\phi_{\text{GEODETIC}}) \times \left\{ h + \frac{b^2}{\sqrt{a^2 \cos^2(\phi_{\text{GEODETIC}}) + b^2 \sin^2(\phi_{\text{GEODETIC}})}} \right\}. \tag{B.50}$$

in terms of geodetic latitude ϕ_{GEODETIC}, longitude θ, and orthometric altitude h with respect to the reference geoid.

The inverse transformation, from ECEF XYZ to geodetic longitude–latitude–altitude coordinates, is

$$\theta = \text{a tan 2}(y_{\text{ECEF}}, x_{\text{ECEF}}) \tag{B.51}$$

$$\phi_{\text{GEODETIC}} = \text{a tan 2}\left(z_{\text{ECEF}} + e^2 a^2 \sin^3(\zeta)/b, \xi - e^2 a \cos^3(\zeta)\right) \tag{B.52}$$

$$h = \frac{\xi}{\cos(\phi_{\text{GEODETIC}}) - R_{\text{T}}}, \tag{B.53}$$

where

$$\zeta = \text{atan2}(az_{\text{ECEF}}, b\xi) \tag{B.54}$$

$$\xi = \sqrt{x_{\text{ECEF}}^2 + y_{\text{ECEF}}^2} \tag{B.55}$$

$$R_{\text{T}} = \frac{a}{\sqrt{1 - e^2 \sin^2(\phi)}}, \tag{B.56}$$

where R_{T} is the transverse radius of curvature on the ellipsoid, a is the equatorial radius, b is the polar radius, and e is elliptical eccentricity.

B.3.5.6 WGS84 Reference Geoid Parameters There are several reference geoids used throughout the world, each with its own set of parameters. The one commonly used throughout the book has the following parameters:

```
Semi-Major Axis (Equatorial Radius)..6378137.0.............meters
Semi-Minor Axis (Polar Radius).......6356752.3142..........meters
Flattening...........................0.0033528106718309896...unitless
Inverse Flattening...................298.2572229328697......unitless
First Eccentricity...................0.08181919092890624....unitless
First Eccentricity Squared...........0.006694380004260827...unitless
Second Eccentricity..................0.08209443803685366....unitless
Second Eccentricity Squared..........0.006739496756586903...unitless
```

B.3.5.7 Geocentric Latitude For points on the geoid surface, the tangent of geocentric latitude, is the ratio of distance above (+) or below (–) the equator ($z = b \sin(\phi_{\text{PARAMETRIC}})$) to the distance from the polar axis ($x_{\text{MERIDIONAL}} = a \cos(\phi_{\text{PARAMETRIC}})$), or

$$\tan(\phi_{\text{GEOCENTRIC}}) = \frac{b \sin(\phi_{\text{PARAMETRIC}})}{a \cos(\phi_{\text{PARAMETRIC}})} \tag{B.57}$$

$$= \frac{b}{a} \tan(\phi_{\text{PARAMETRIC}}) \tag{B.58}$$

$$= \frac{b^2}{a^2} \tan(\phi_{\text{GEODETIC}}), \tag{B.59}$$

from which, using the same trigonometric identities as were used for geodetic latitude,

$$\sin(\phi_{\text{GEOCENTRIC}}) = \frac{\tan(\phi_{\text{GEOCENTRIC}})}{\sqrt{1 + \tan^2(\phi_{\text{GEOCENTRIC}})}} \tag{B.60}$$

$$= \frac{b \sin(\phi_{\text{PARAMETRIC}})}{\sqrt{a^2 \cos^2(\phi_{\text{PARAMETRIC}}) + b^2 \sin^2(\phi_{\text{PARAMETRIC}})}} \tag{B.61}$$

$$= \frac{b^2 \sin(\phi_{\text{GEODETIC}})}{\sqrt{a^4 \cos^2(\phi_{\text{GEODETIC}}) + b^4 \sin^2(\phi_{\text{GEODETIC}})}} \tag{B.62}$$

$$\cos(\phi_{\text{GEOCENTRIC}}) = \frac{1}{\sqrt{1 + \tan^2(\phi_{\text{GEOCENTRIC}})}} \tag{B.63}$$

$$= \frac{a \cos(\phi_{\text{PARAMETRIC}})}{\sqrt{a^2 \cos^2(\phi_{\text{PARAMETRIC}}) + b^2 \sin^2(\phi_{\text{PARAMETRIC}})}} \tag{B.64}$$

$$= \frac{a^2 \cos(\phi_{\text{GEODETIC}})}{\sqrt{a^4 \cos^2(\phi_{\text{GEODETIC}}) + b^4 \sin^2(\phi_{\text{GEODETIC}})}}. \tag{B.65}$$

The inverse relationships are

$$\tan(\phi_{\text{PARAMETRIC}}) = \frac{a}{b} \tan(\phi_{\text{GEOCENTRIC}}) \tag{B.66}$$

$$\tan(\phi_{\text{GEODETIC}}) = \frac{a^2}{b^2} \tan(\phi_{\text{GEOCENTRIC}}), \tag{B.67}$$

from which, using the same trigonometric identities again,

$$\sin(\phi_{\text{PARAMETRIC}}) = \frac{\tan(\phi_{\text{PARAMETRIC}})}{\sqrt{1 + \tan^2(\phi_{\text{PARAMETRIC}})}} \tag{B.68}$$

$$= \frac{a \sin(\phi_{\text{GEOCENTRIC}})}{\sqrt{a^2 \sin^2(\phi_{\text{GEOCENTRIC}}) + b^2 \cos^2(\phi_{\text{GEOCENTRIC}})}} \tag{B.69}$$

$$\sin(\phi_{\text{GEODETIC}}) = \frac{a^2 \sin(\phi_{\text{GEOCENTRIC}})}{\sqrt{a^4 \sin^2(\phi_{\text{GEOCENTRIC}}) + b^4 \cos^2(\phi_{\text{GEOCENTRIC}})}} \tag{B.70}$$

$$\cos(\phi_{\text{PARAMETRIC}}) = \frac{1}{\sqrt{1 + \tan^2(\phi_{\text{PARAMETRIC}})}} \tag{B.71}$$

$$= \frac{b \cos(\phi_{\text{GEOCENTRIC}})}{\sqrt{a^2 \sin^2(\phi_{\text{GEOCENTRIC}}) + b^2 \cos^2(\phi_{\text{GEOCENTRIC}})}} \tag{B.72}$$

$$\cos(\phi_{\text{GEODETIC}}) = \frac{b^2 \cos(\phi_{\text{GEOCENTRIC}})}{\sqrt{a^4 \sin^2(\phi_{\text{GEOCENTRIC}}) + b^4 \cos^2(\phi_{\text{GEOCENTRIC}})}}. \tag{B.73}$$

B.3.5.8 Geocentric Radius Geocentric radius $R_{\text{GEOCENTRIC}}$ is the distance to the center of the earth. As a function of geodetic latitude ϕ_{GEODETIC},

$$R_{\text{GEOCENTRIC}}(\phi_{\text{GEODETIC}}) = \sqrt{\frac{\left(a^2 \cos\phi_{\text{GEODETIC}}\right)^2 + \left(b^2 \sin\phi_{\text{GEODETIC}}\right)^2}{\left(a \cos\phi_{\text{GEODETIC}}\right)^2 + \left(b \sin\phi_{\text{GEODETIC}}\right)^2}}. \tag{B.74}$$

B.3.6 Ellipsoidal Radius of Curvature

The radius of curvature on the reference ellipsoidal surface is what determines how geodetic longitude and latitude change with distance traveled over the surface. It is generally different in different directions, except at the poles. At other places, it is specified by two different values, corresponding to two directions.

Meridional radius of curvature is measured in the north–south direction[2]:

$$R_{\mathrm{M}} = \frac{(ab)^2}{\left[(a\cos\phi_{\mathrm{GEODETIC}})^2 + (b\sin\phi_{\mathrm{GEODETIC}}]^2\right)^{3/2}}. \tag{B.75}$$

Transverse radius of curvature is measured in the east–west direction. It is defined by the angular rate of the local vertical about the north direction as a function of east velocity:

$$\omega_{\mathrm{N}} = \frac{v_{\mathrm{E}}}{R_{\mathrm{T}}} \tag{B.76}$$

$$R_{\mathrm{T}} = \frac{a^2}{\sqrt{(a\cos\phi_{\mathrm{GEODETIC}})^2 + (b\sin\phi_{\mathrm{GEODETIC}})^2}}. \tag{B.77}$$

B.3.7 Local Tangent Plane (LTP) Coordinates

LTP coordinates, also called "locally level coordinates," are a return to the first-order model of the earth as being flat, where they serve as local reference directions for representing vehicle attitude and velocity for operation on or near the surface of the earth. A common orientation for LTP coordinates has one horizontal axis (the north axis) in the direction of increasing latitude and the other horizontal axis (the east axis) in the direction of increasing longitude—as illustrated in Fig. B.7.

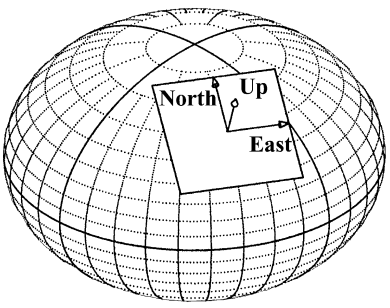


Fig. B.7 ENU coordinates.

[2]The direction measured ~240 BCE by the Greek polymath Aristophanes (~276–195 BCE).

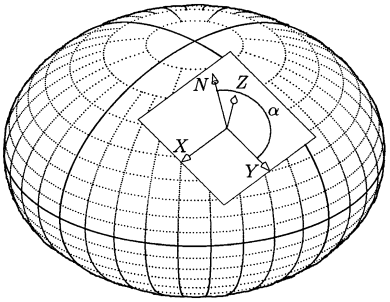


Fig. B.8 Alpha wander coordinates.

Horizontal location components in this local coordinate frame are called "relative northing" and "relative easting."

B.3.7.1 Alpha Wander Coordinates Maintaining east–north orientation was a problem for some INSs at the poles, where north and east directions change by 180°. Early gimbaled inertial systems could not slew the platform axes fast enough for near-polar operation. This problem was solved by letting the platform axes "wander" from north, but keeping track of the angle α between north and a reference platform axis, as shown in Fig. B.8. This LTP orientation came to be called "alpha wander."

B.3.7.2 ENU/NED Coordinates ENU and NED are two common right-handed LTP coordinate systems. ENU coordinates may be preferred to NED coordinates because altitude increases in the upward direction. But NED coordinates may also be preferred over ENU coordinates because the direction of a right (clockwise) turn is in the positive direction with respect to a downward axis, and NED coordinate axes coincide with vehicle-fixed RPY coordinates (Section B.3.8) when the vehicle is level and headed north.

The coordinate transformation matrix $\mathbf{C}_{\text{NED}}^{\text{ENU}}$ from ENU to NED coordinates and the transformation matrix $\mathbf{C}_{\text{ENU}}^{\text{NED}}$ from NED to ENU coordinates are one and the same:

$$\mathbf{C}_{\text{NED}}^{\text{ENU}} = \mathbf{C}_{\text{ENU}}^{\text{NED}} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{B.78}$$

B.3.7.3 ENU/ECEF Coordinates The unit vectors in local *east*, *north*, and *up* directions, as expressed in ECEF Cartesian coordinates, will be

$$\vec{\mathbf{1}}_{\mathrm{E}} = \begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \\ 0 \end{bmatrix} \tag{B.79}$$

$$\vec{\mathbf{1}}_{\mathrm{N}} = \begin{bmatrix} -\cos(\theta)\sin\left(\phi_{\mathrm{geodetic}}\right) \\ -\sin(\theta)\sin\left(\phi_{\mathrm{geodetic}}\right) \\ \cos\left(\phi_{\mathrm{geodetic}}\right) \end{bmatrix} \tag{B.80}$$

$$\vec{\mathbf{1}}_{\mathrm{U}} = \begin{bmatrix} \cos(\theta)\cos\left(\phi_{\mathrm{geodetic}}\right) \\ \sin(\theta)\cos\left(\phi_{\mathrm{geodetic}}\right) \\ \sin\left(\phi_{\mathrm{geodetic}}\right) \end{bmatrix}, \tag{B.81}$$

and the unit vectors in the ECEF X, Y, and Z directions, as expressed in ENU coordinates, will be

$$\vec{\mathbf{1}}_{X} = \begin{bmatrix} -\sin(\theta) \\ -\cos(\theta)\sin\left(\phi_{\mathrm{geodetic}}\right) \\ \cos(\theta)\cos\left(\phi_{\mathrm{geodetic}}\right) \end{bmatrix} \tag{B.82}$$

$$\vec{\mathbf{1}}_{Y} = \begin{bmatrix} \cos(\theta) \\ -\sin(\theta)\sin\left(\phi_{\mathrm{geodetic}}\right) \\ \sin(\theta)\cos\left(\phi_{\mathrm{geodetic}}\right) \end{bmatrix} \tag{B.83}$$

$$\vec{\mathbf{1}}_{Z} = \begin{bmatrix} 0 \\ \cos\left(\phi_{\mathrm{geodetic}}\right) \\ \sin\left(\phi_{\mathrm{geodetic}}\right) \end{bmatrix}. \tag{B.84}$$

B.3.7.4 NED/ECEF Coordinates It is more natural in some applications to use NED directions for locally level coordinates. This coordinate system coincides with vehicle-body-fixed RPY coordinates (shown in Fig. B.9) when the vehicle is level headed north. The unit vectors in local *north*, *east*, and *down* directions, as expressed in ECEF Cartesian coordinates, will be

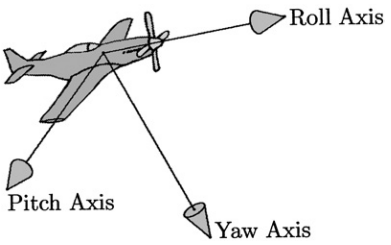


Fig. B.9 Roll–pitch–yaw axes.

$$\vec{\mathbf{1}}_{\mathrm{N}} = \begin{bmatrix} -\cos(\theta)\sin\left(\phi_{\text{geodetic}}\right) \\ -\sin(\theta)\sin\left(\phi_{\text{geodetic}}\right) \\ \cos\left(\phi_{\text{geodetic}}\right) \end{bmatrix} \tag{B.85}$$

$$\vec{\mathbf{1}}_{\mathrm{E}} = \begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \\ 0 \end{bmatrix} \tag{B.86}$$

$$\vec{\mathbf{1}}_{\mathrm{D}} = \begin{bmatrix} -\cos(\theta)\cos\left(\phi_{\text{geodetic}}\right) \\ -\sin(\theta)\cos\left(\phi_{\text{geodetic}}\right) \\ -\sin\left(\phi_{\text{geodetic}}\right) \end{bmatrix}, \tag{B.87}$$

and the unit vectors in the ECEF X, Y, and Z directions, as expressed in NED coordinates, will be

$$\vec{\mathbf{1}}_{X} = \begin{bmatrix} -\cos(\theta)\sin\left(\phi_{\text{geodetic}}\right) \\ -\sin(\theta) \\ -\cos(\theta)\cos\left(\phi_{\text{geodetic}}\right) \end{bmatrix} \tag{B.88}$$

$$\vec{\mathbf{1}}_{Y} = \begin{bmatrix} -\sin(\theta)\sin\left(\phi_{\text{geodetic}}\right) \\ \cos(\theta) \\ -\sin(\theta)\cos\left(\phi_{\text{geodetic}}\right) \end{bmatrix} \tag{B.89}$$

$$\vec{\mathbf{1}}_{Z} = \begin{bmatrix} \cos\left(\phi_{\text{geodetic}}\right) \\ 0 \\ -\sin\left(\phi_{\text{geodetic}}\right) \end{bmatrix}. \tag{B.90}$$

B.3.8 RPY Coordinates

RPY coordinates are vehicle fixed, with the roll axis in the nominal direction of motion of the vehicle, the pitch axis out the right-hand side, and the yaw axis such that turning to the right is positive, as illustrated in Fig. B.9. The same orientations of vehicle-fixed coordinates are used for surface ships and ground vehicles. They are also called "SAE coordinates" because they are the standard body-fixed coordinates used by the Society of Automotive Engineers.

For rocket boosters with their roll axes vertical at liftoff, the pitch axis is typically defined to be orthogonal to the plane of the boost trajectory (also called the "pitch plane" or "ascent plane").

B.3.9 Vehicle Attitude Euler Angles

The attitude of the vehicle body with respect to local coordinates can be specified in terms of rotations about the vehicle roll, pitch, and yaw axes, starting with these axes aligned with NED coordinates. The angles of rotation about

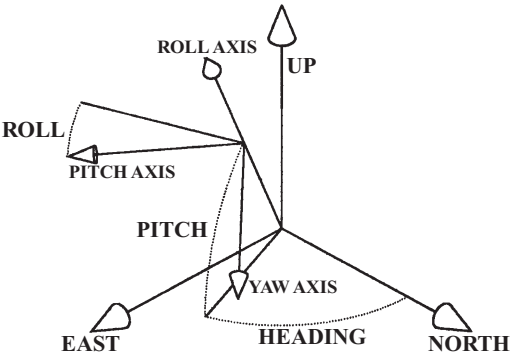


Fig. B.10 Vehicle Euler angles.

each of these axes are called *Euler angles*, named for the Swiss mathematician Leonard Euler (1707–1783). It is always necessary to specify the order of rotations when specifying Euler (rhymes with "oiler") angles.

A fairly common convention for vehicle attitude Euler angles is illustrated in Fig. B.10, where, starting with the vehicle level with roll axis pointed north,

1. *Yaw/Heading.* Rotate through the yaw angle (Y) about the vehicle yaw axis to the intended azimuth (heading) of the vehicle roll axis. Azimuth is measured clockwise (east) from north.
2. *Pitch.* Rotate through the pitch angle (P) about the vehicle pitch axis to bring the vehicle roll axis to its intended elevation. Elevation is measured positive upward from the local horizontal plane.
3. *Roll.* Rotate through the roll angle (R) about the vehicle roll axis to bring the vehicle attitude to the specified orientation.

Euler angles are redundant for vehicle attitudes with 90° pitch, in which case the roll axis is vertical. In that attitude, heading changes also rotate the vehicle about the roll axis. This is the attitude of most rocket boosters at liftoff. Some boosters can be seen making a roll maneuver immediately after liftoff to align their yaw axes with the launch azimuth in the ascent plane. This maneuver may be required to correct for launch delays on missions for which launch azimuth is a function of launch time.

B.3.9.1 RPY/ENU Coordinates With vehicle attitude specified by yaw angle (Y), pitch angle (P), and roll angle (R) as specified above, the resulting unit vectors of the roll, pitch, and yaw axes in ENU coordinates will be

$$\vec{\mathbf{1}}_{\mathrm{R}} = \begin{bmatrix} \sin(Y)\cos(P) \\ \cos(Y)\cos(P) \\ \sin(P) \end{bmatrix} \tag{B.91}$$

$$\vec{\mathbf{1}}_{\mathrm{P}} = \begin{bmatrix} \cos(R)\cos(Y)+\sin(R)\sin(Y)\sin(P) \\ -\cos(R)\sin(Y)+\sin(R)\cos(Y)\sin(P) \\ -\sin(R)\cos(P) \end{bmatrix} \tag{B.92}$$

$$\vec{\mathbf{1}}_{\mathrm{Y}} = \begin{bmatrix} -\sin(R)\cos(Y)+\cos(R)\sin(Y)\sin(P) \\ \sin(R)\sin(Y)+\cos(R)\cos(Y)\sin(P) \\ -\cos(R)\cos(P) \end{bmatrix}; \tag{B.93}$$

the unit vectors of the east, north, and up axes in RPY coordinates will be

$$\vec{\mathbf{1}}_{\mathrm{E}} = \begin{bmatrix} \sin(Y)\cos(P) \\ \cos(R)\cos(Y)+\sin(R)\sin(Y)\sin(P) \\ -\sin(R)\cos(\boldsymbol{Y})+\cos(R)\sin(Y)\sin(P) \end{bmatrix} \tag{B.94}$$

$$\vec{\mathbf{1}}_{\mathrm{N}} = \begin{bmatrix} \cos(Y)\cos(P) \\ -\cos(R)\sin(Y)+\sin(R)\cos(Y)\sin(P) \\ \sin(R)\sin(\boldsymbol{Y})+\cos(R)\cos(Y)\sin(P) \end{bmatrix} \tag{B.95}$$

$$\vec{\mathbf{1}}_{\mathrm{U}} = \begin{bmatrix} \sin(P) \\ -\sin(R)\cos(P) \\ -\cos(R)\cos(P) \end{bmatrix}; \tag{B.96}$$

and the coordinate transformation matrix from RPY coordinates to ENU coordinates will be

$$\mathbf{C}_{\mathrm{ENU}}^{\mathrm{RPY}} = \begin{bmatrix} \vec{\mathbf{1}}_R & \vec{\mathbf{1}}_P & \vec{\mathbf{1}}_Y \end{bmatrix} = \begin{bmatrix} \vec{\mathbf{1}}_E^T \\ \vec{\mathbf{1}}_N^T \\ \vec{\mathbf{1}}_U^T \end{bmatrix} \tag{B.97}$$

$$= \begin{bmatrix} S_{\mathrm{Y}}C_{\mathrm{P}} & C_{\mathrm{R}}C_{\mathrm{Y}}+S_{\mathrm{R}}S_{\mathrm{Y}}S_{\mathrm{P}} & -S_{\mathrm{R}}C_{\mathrm{Y}}+C_{\mathrm{R}}S_{\mathrm{Y}}S_{\mathrm{P}} \\ C_{\mathrm{Y}}C_{\mathrm{P}} & -C_{\mathrm{R}}S_{\mathrm{Y}}+S_{\mathrm{R}}C_{\mathrm{Y}}S_{\mathrm{P}} & S_{\mathrm{R}}S_{\mathrm{Y}}+C_{\mathrm{R}}C_{\mathrm{Y}}S_{\mathrm{P}} \\ S_{\mathrm{P}} & -S_{\mathrm{R}}C_{\mathrm{P}} & -C_{\mathrm{R}}C_{\mathrm{P}} \end{bmatrix}, \tag{B.98}$$

where

$$S_{\mathrm{R}} = \sin(R) \tag{B.99}$$

$$C_{\mathrm{R}} = \cos(R) \tag{B.100}$$

$$S_{\mathrm{P}} = \sin(P) \tag{B.101}$$

$$C_{\mathrm{P}} = \cos(P) \tag{B.102}$$

$$S_{\mathrm{Y}} = \sin(Y) \tag{B.103}$$

$$C_{\mathrm{Y}} = \cos(Y). \tag{B.104}$$

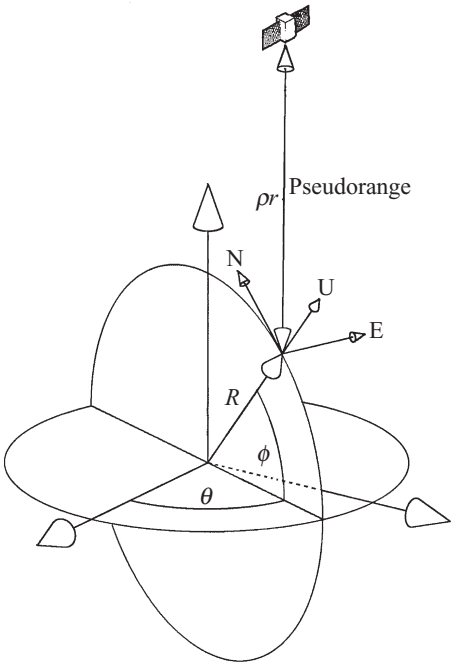


Fig. B.11 Pseudorange between satellite and observer.

B.3.10 GNSS Navigation Coordinates

The principal coordinate systems used in GNSS navigation in the terrestrial environment are shown in Figs. B.7 (ENU, for local terrestrial navigation), B.11 (ECEF, for global terrestrial navigation), and B.12 (satellite orbital position with respect to polar ECI coordinates).

The parameter Ω in Fig. B.12 is the right ascension of the ascending node (RAAN), the ECI longitude where the orbital plane intersects the equatorial plane as the satellite crosses from the southern hemisphere to the northern hemisphere. The orbital plane is specified by Ω and α, the inclination of the orbit plane with respect to the equatorial plane ($\alpha \approx 55^\circ$ for GPS satellite orbits). The θ_{SAT} parameter in the figure represents the location of the satellite within the orbit plane, as the angular phase in the circular orbit with respect to ascending node.

For GNSS satellite orbits, the satellite angle θ_{SAT} changes at a nearly constant rate of about 1.4584×10^{-4} rad/s and for a period of about 43,082 s (half a day).

The nominal satellite position in ECEF coordinates is then

$$x = R[\cos\theta_{SAT}\cos\Omega - \sin\theta_{SAT}\sin\Omega\cos\alpha] \tag{B.105}$$

$$y = R[\cos\theta_{SAT}\sin\Omega + \sin\theta_{SAT}\cos\Omega\cos\alpha] \tag{B.106}$$

$$z = R\sin\theta_{SAT}\sin\alpha \tag{B.107}$$

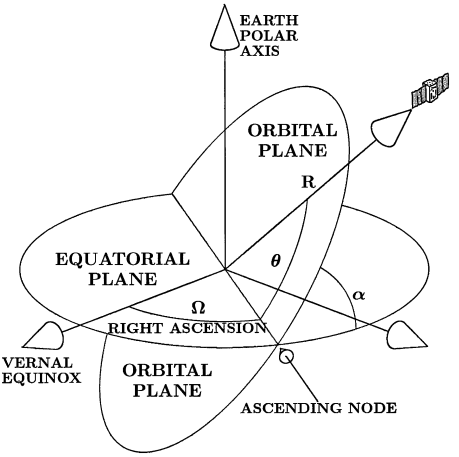


Fig. B.12 Satellite orbital coordinates.

$$\theta_{\text{SAT}} = \theta_0 + (t - t_0)\frac{360}{43{,}082}\,\text{deg} \tag{B.108}$$

$$\Omega = \Omega_0 - (t - t_0)\frac{360}{86{,}164}\,\text{deg} \tag{B.109}$$

$$R = 26{,}560{,}000\ \text{m}. \tag{B.110}$$

GNSS satellite positions in the transmitted navigation message are typically specified in the ECEF Coordinate System of WGS84. Additionally, a locally level east-north-up (ENU) reference coordinate system (described in Section B.3.7) is used by an observer located on Earth (see Fig. B.7):

$$X_{\text{ENU}} = \mathbf{C}_{\text{ENU}}^{\text{ECEF}} X_{\text{ECEF}} + S_{\text{ENU}}$$

$\mathbf{C}_{\text{ENU}}^{\text{ECEF}}$ = coordinate transformation matrix from ECEF to ENU

S_{ENU} = coordinate origin shift vector from ECEF to local ENU, in ENU coordinates

$$\mathbf{C}_{\text{ENU}}^{\text{ECEF}} = \begin{bmatrix} -\sin\theta & \cos\theta & 0 \\ -\sin\phi\cos\theta & -\sin\phi\sin\theta & \cos\phi \\ \cos\phi\cos\theta & \cos\phi\sin\theta & \sin\phi \end{bmatrix}$$

$$S_{\text{ENU}} = \begin{bmatrix} X_{\text{U}}\sin\theta - Y_{\text{U}}\cos\theta \\ X_{\text{U}}\sin\phi\cos\theta - Y_{\text{U}}\sin\phi\sin\theta - Z_{\text{U}}\cos\phi \\ -X_{\text{U}}\cos\phi\cos\theta - Y_{\text{U}}\cos\phi\sin\theta - Z_{\text{U}}\sin\theta \end{bmatrix}$$

$X_{\text{U}}, Y_{\text{U}}, Z_{\text{U}}$ = user's position in ECEF coordinates

θ = local geodetic longitude

ϕ = local geodetic latitude.

B.4 COORDINATE TRANSFORMATION MODELS

Coordinate transformations are methods for transforming a vector represented in one coordinate system into the appropriate representation in another coordinate system. These coordinate transformations can be represented in a number of different ways, each with its advantages and disadvantages.

These transformations generally involve translations (for coordinate systems with different origins) and rotations (for Cartesian coordinate systems with different axis directions) or transcendental transformations (between Cartesian and polar or geodetic coordinates). The transformations between Cartesian and polar coordinates have already been discussed in Section B.3.1 and translations are rather obvious, so we will concentrate on the rotations.

B.4.1 Euler Angles

Euler (rhymes with "oiler") angles were used for defining vehicle attitude in Section B.3.9, and vehicle attitude representation is a common use of Euler angles in navigation.

Euler angles are used to define a coordinate transformation in terms of a set of three angular rotations, performed in a specified sequence about three specified orthogonal axes, to bring one coordinate frame to coincide with another. The coordinate transformation from RPY coordinates to NED coordinates, for example, can be composed from three Euler rotation matrices

$$\mathbf{C}_{\text{NED}}^{\text{RPY}} = \overbrace{\begin{bmatrix} C_Y & -S_Y & 0 \\ S_Y & C_Y & 0 \\ 0 & 0 & 1 \end{bmatrix}}^{\text{YAW}} \overbrace{\begin{bmatrix} C_P & 0 & S_P \\ 0 & 1 & 0 \\ -S_P & 0 & C_P \end{bmatrix}}^{\text{PITCH}} \overbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & C_R & -S_R \\ 0 & S_R & C_R \end{bmatrix}}^{\text{ROLL}}. \tag{B.111}$$

$$= \underbrace{\begin{bmatrix} C_Y C_P & -S_Y C_R + C_Y S_P S_R & S_Y S_R + C_Y S_P C_R \\ S_Y C_P & C_Y C_R + S_Y S_P S_R & -C_Y S_R + S_Y S_P C_R \\ -S_P & C_P S_R & C_P C_R \\ \text{(roll axis)} & \text{(pitch axis)} & \text{(yaw axis)} \end{bmatrix}}_{\text{in NED coordinates}}, \tag{B.112}$$

where the matrix elements are defined in Eq. B.99–B.104. This matrix also rotates the NED coordinate axes to coincide with RPY coordinate axes. (Compare this with the transformation from RPY to ENU coordinates in Eq. B.98.)

For example, the coordinate transformation for nominal booster rocket launch attitude (roll axis straight up) would be given by Eq. B.112 with pitch angle $P = \pi/2$ ($C_P = 0$, $S_P = 1$), which becomes

$$\mathbf{C}_{\text{NED}}^{\text{RPY}} = \begin{bmatrix} 0 & \sin(R-Y) & \cos(R-Y) \\ 0 & \cos(R-Y) & -\sin(R-Y) \\ 1 & 0 & 0 \end{bmatrix};$$

that is, the coordinate transformation in this attitude depends only on the difference between roll angle (R) and yaw angle (Y). Euler angles are a concise representation for vehicle attitude. They are handy for driving cockpit displays such as compass cards (using Y) and artificial horizon indicators (using R and P), but they are not particularly handy for representing vehicle attitude dynamics. The reasons for the latter include

- Euler angles have discontinuities analogous to "gimbal lock" when the vehicle roll axis is pointed upward, as it is for launch of many rockets. In that orientation, tiny changes in vehicle pitch or yaw cause ±180° changes in heading angle. For aircraft, this creates a slewing rate problem for electromechanical compass card displays.
- The relationships between sensed body rates and Euler angle rates are mathematically complicated.

B.4.2 Rotation Vectors

All right-handed orthogonal coordinate systems with the same origins in three dimensions can be transformed one onto another by single rotations about fixed axes. The corresponding **rotation vectors** relating two coordinate systems are defined by the direction (rotation axis) and magnitude (rotation angle) of that transformation.

For example, the rotation vector for rotating ENU coordinates to NED coordinates (and vice versa) is

$$\vec{\boldsymbol{\rho}}_{\text{NED}}^{\text{ENU}} = \begin{bmatrix} \pi/\sqrt{2} \\ \pi/\sqrt{2} \\ 0 \end{bmatrix}, \tag{B.113}$$

which has magnitude $\left|\vec{\boldsymbol{\rho}}_{\text{NED}}^{\text{ENU}}\right| = \pi$ (180°) and direction north–east, as illustrated in Fig. B.13. In transforming between ENU coordinates to NED coordinates, note that the transformation represents a change in coordinates, not a physical rotation of anything else. North is still north, but changes between being the second component to the first component. Up is still up, it is just that down is now the direction of the third component.

The rotation vector is another minimal representation of a coordinate transformation, along with Euler angles. Like Euler angles, rotation vectors are concise but also have some drawbacks:

1. It is not a unique representation, in that adding multiples of $\pm 2\pi$ to the magnitude of a rotation vector has no effect on the transformation it represents.
2. It is a nonlinear and rather complicated representation, in that the result of one rotation followed by another is a third rotation, the rotation

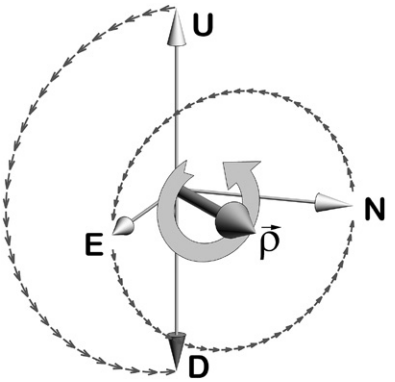


Fig. B.13 Rotation from ENU to NED coordinates.

vector for which is a fairly complicated function of the first two rotation vectors.

But, unlike Euler angles, rotation vector models do not exhibit "gimbal lock."

B.4.2.1 Rotation Vector to Matrix The rotation represented by a rotation vector

$$\vec{\boldsymbol{\rho}} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \end{bmatrix} \tag{B.114}$$

can be implemented as multiplication by the matrix

$$\mathbf{C}(\vec{\boldsymbol{\rho}}) \overset{\text{def}}{=} \exp(\vec{\boldsymbol{\rho}}\otimes) \tag{B.115}$$

$$\overset{\text{def}}{=} \exp\left(\begin{bmatrix} 0 & -\rho_3 & \rho_2 \\ \rho_3 & 0 & -\rho_1 \\ -\rho_2 & \rho_1 & 0 \end{bmatrix}\right) \tag{B.116}$$

$$= \cos(|\vec{\boldsymbol{\rho}}|)\mathbf{I}_3 + \frac{1-\cos(|\vec{\boldsymbol{\rho}}|)}{|\vec{\boldsymbol{\rho}}|^2}\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^T + \frac{\sin(|\vec{\boldsymbol{\rho}}|)}{|\vec{\boldsymbol{\rho}}|}\begin{bmatrix} 0 & -\rho_3 & \rho_2 \\ \rho_3 & 0 & -\rho_1 \\ -\rho_2 & \rho_1 & 0 \end{bmatrix}. \tag{B.117}$$

$$= \cos(\theta)\mathbf{I}_3 + (1-\cos(\theta))\vec{\mathbf{1}}_\rho\vec{\mathbf{1}}_\rho^T + \sin(\theta)\begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \tag{B.118}$$

$$\theta \stackrel{\text{def}}{=} |\vec{\boldsymbol{\rho}}| \tag{B.119}$$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \stackrel{\text{def}}{=} \frac{\vec{\boldsymbol{\rho}}}{|\vec{\boldsymbol{\rho}}|}; \tag{B.120}$$

that is, for any three-rowed column vector $\mathbf{v}$, $\mathbf{C}(\vec{\boldsymbol{\rho}})\mathbf{v}$ rotates it through an angle of $|\vec{\boldsymbol{\rho}}|$ radians about the vector $\vec{\boldsymbol{\rho}}$.

The form of the matrix in Eq. B.118[3] is better suited for computation when $\theta \approx 0$, but the form of the matrix in Eq. B.117 is useful for computing sensitivities using partial derivatives.

For example, the rotation vector $\vec{\boldsymbol{\rho}}_{\text{NED}}^{\text{ENU}}$ in Eq. B.113 transforming between ENU and NED has magnitude and direction,

$$\theta = \pi \quad (\sin(\theta) = 0, \cos(\theta) = -1)$$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix},$$

respectively, and the corresponding rotation matrix

$$\begin{aligned}
\mathbf{C}_{\text{NED}}^{\text{ENU}} &= \cos(\pi)\mathbf{I}_3 + (1-\cos(\pi))\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}^T + \sin(\pi)\begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \\
&= -\mathbf{I}_3 + 2\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}^T + 0 \\
&= \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}
\end{aligned}$$

transforms from ENU to NED coordinates. (Compare this result to Eq. B.78.) Because coordinate transformation matrices are orthogonal matrices and the matrix $\mathbf{C}_{\text{NED}}^{\text{ENU}}$ is also symmetric, $\mathbf{C}_{\text{NED}}^{\text{ENU}}$ is its own inverse; that is,

[3]Linear combinations of the sort $a_1\mathbf{I}_{3\times3} + a_2\left[\vec{\mathbf{1}}_\rho\otimes\right] + a_3\vec{\mathbf{1}}_\rho\vec{\mathbf{1}}_\rho^T$, where $\mathbf{u}$ is a unit vector, form a subalgebra of 3×3 matrices with relatively simple rules for multiplication, inversion, and so on.

$$\mathbf{C}_{\text{NED}}^{\text{ENU}} = \mathbf{C}_{\text{ENU}}^{\text{NED}}. \tag{B.121}$$

B.4.2.2 Matrix to Rotation Vector Although there is a unique coordinate transformation matrix for each rotation vector, the converse is not true. Adding multiples of 2π to the magnitude of a rotation vector has no effect on the resulting coordinate transformation matrix. The following approach yields a unique rotation vector with magnitude $|\vec{\rho}| \leq \pi$.

The trace $\text{tr}(\mathbf{C})$ of a square matrix $\mathbf{M}$ is the sum of its diagonal values. For the coordinate transformation matrix of Eq. B.117,

$$\text{tr}(\mathbf{C}(\vec{\rho})) = 1 + 2\cos(\theta), \tag{B.122}$$

from which the rotation angle

$$|\vec{\rho}| = \theta \tag{B.123}$$

$$= \arccos\left(\frac{\text{tr}(\mathbf{C}(\vec{\rho})) - 1}{2}\right), \tag{B.124}$$

a formula which will yield a result in the range $0 < \theta < \pi$, but with poor fidelity near where the derivative of the cosine equals zero at $\theta = 0$ and $\theta = \pi$.

The values of θ near $\theta = 0$ and $\theta = \pi$ can be better estimated using the sine of θ, which can be recovered using the antisymmetric part of $\mathbf{C}(\vec{\rho})$,

$$\mathbf{A} = \begin{bmatrix} 0 & -a_{21} & a_{13} \\ a_{21} & 0 & -a_{32} \\ -a_{13} & a_{32} & 0 \end{bmatrix} \tag{B.125}$$

$$\overset{\text{def}}{=} \frac{1}{2}\left[\mathbf{C}(\vec{\rho}) - \mathbf{C}^T(\vec{\rho})\right] \tag{B.126}$$

$$= \frac{\sin(\theta)}{\theta}\begin{bmatrix} 0 & -\rho_3 & \rho_2 \\ \rho_3 & 0 & -\rho_1 \\ -\rho_2 & \rho_1 & 0 \end{bmatrix}, \tag{B.127}$$

from which the vector

$$\begin{bmatrix} a_{32} \\ a_{13} \\ a_{21} \end{bmatrix} = \sin(\theta)\frac{1}{|\vec{\rho}|}\vec{\rho} \tag{B.128}$$

will have magnitude

$$\sqrt{a_{32}^2 + a_{13}^2 + a_{21}^2} = \sin(\theta) \tag{B.129}$$

and the same direction as $\vec{\rho}$. As a consequence, one can recover the magnitude θ of $\vec{\rho}$ from

$$\theta = \text{atan2}\left(\sqrt{a_{32}^2 + a_{13}^2 + a_{21}^2}, \frac{\text{tr}(\mathbf{C}(\vec{\rho})) - 1}{2} \right), \tag{B.130}$$

using the MATLAB® function atan2, and then the rotation vector $\vec{\rho}$ as

$$\vec{\rho} = \frac{\theta}{\sin(\theta)} \begin{bmatrix} a_{32} \\ a_{13} \\ a_{21} \end{bmatrix} \tag{B.131}$$

when $0 < \theta < \pi$.

B.4.2.3 Special Cases for sin(θ) ≈ 0 For $\theta \approx 0$, $\vec{\rho} \approx 0$, although Eq. B.131 may still work adequately for $\theta > 10^{-6}$, say.

For $\theta \approx \pi$, the symmetric part of $\mathbf{C}(\vec{\rho})$,

$$\mathbf{S} = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{12} & s_{22} & s_{23} \\ s_{13} & s_{23} & s_{33} \end{bmatrix} \tag{B.132}$$

$$\stackrel{\text{def}}{=} \frac{1}{2}\left[\mathbf{C}(\vec{\rho}) + \mathbf{C}^T(\vec{\rho})\right] \tag{B.133}$$

$$= \cos(\theta)\mathbf{I}_3 + \frac{1 - \cos(\theta)}{\theta^2}\vec{\rho}\vec{\rho}^T \tag{B.134}$$

$$\approx -\mathbf{I}_3 + \frac{2}{\theta^2}\vec{\rho}\vec{\rho}^T, \tag{B.135}$$

and the unit vector

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \stackrel{\text{def}}{=} \frac{1}{\theta}\vec{\rho} \tag{B.136}$$

satisfies

$$\mathbf{S} \approx \begin{bmatrix} 2u_1^2 - 1 & 2u_1u_2 & 2u_1u_3 \\ 2u_1u_2 & 2u_2^2 - 1 & 2u_2u_3 \\ 2u_1u_3 & 2u_2u_3 & 2u_3^2 - 1 \end{bmatrix}, \tag{B.137}$$

which can be solved for a unique u by assigning $u_k > 0$ for

$$k = \text{argmax}\left(\begin{bmatrix} s_{11} \\ s_{22} \\ s_{33} \end{bmatrix}\right) \tag{B.138}$$

$$u_k = \sqrt{\frac{s_{kk}+1}{2}}; \tag{B.139}$$

then, depending on whether $k = 1$, $k = 2$ or $k = 3$,

$$\left.\begin{array}{rcccc} & & k=1 & k=2 & k=3 \\ u_1 & \approx & \sqrt{\dfrac{s_{11}+1}{2}} & s_{12}/2u_2 & s_{13}/2u_3 \\ u_2 & \approx & s_{12}/2u_1 & \sqrt{\dfrac{s_{22}+1}{2}} & s_{23}/2u_2 \\ u_3 & \approx & s_{13}/2u_1 & s_{23}/2u_2 & \sqrt{\dfrac{s_{11}+1}{2}} \end{array}\right\} \tag{B.140}$$

and

$$\vec{\boldsymbol{\rho}} = \theta \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}. \tag{B.141}$$

B.4.2.4 Time Derivatives of Rotation Vectors The mathematical relationships between rotation rates ω_k and the time derivatives of the corresponding rotation vector $\vec{\boldsymbol{\rho}}$ are fairly complicated, but they can be derived from Eq. B.226 for the dynamics of coordinate transformation matrices.

Let $\vec{\boldsymbol{\rho}}_{\text{ENU}}$ be the rotation vector represented in earth-fixed ENU coordinates that rotates earth-fixed ENU coordinate axes into vehicle body-fixed RPY axes, and let $\mathbf{C}(\vec{\boldsymbol{\rho}})$ be the corresponding rotation matrix, so that, in ENU coordinates

$$\begin{aligned} \vec{\mathbf{1}}_{\text{E}} &= [1 \quad 0 \quad 0]^{\text{T}} \\ \vec{\mathbf{1}}_{\text{N}} &= [0 \quad 1 \quad 0]^{\text{T}} \\ \vec{\mathbf{1}}_{\text{U}} &= [0 \quad 0 \quad 1]^{\text{T}} \\ \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\vec{\mathbf{1}}_{\text{E}} &= \vec{\mathbf{1}}_{\text{R}} \\ \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\vec{\mathbf{1}}_{\text{N}} &= \vec{\mathbf{1}}_{\text{P}} \\ \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\vec{\mathbf{1}}_{\text{U}} &= \vec{\mathbf{1}}_{\text{Y}} \end{aligned}$$

$$\begin{aligned}\mathbf{C}_{\text{ENU}}^{\text{RPY}} &= \left[\vec{\mathbf{i}}_{\text{R}} \quad \vec{\mathbf{i}}_{\text{P}} \quad \vec{\mathbf{i}}_{\text{Y}}\right] \\ &= \left[\mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\vec{\mathbf{i}}_{\text{E}} \quad \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\vec{\mathbf{i}}_{\text{N}} \quad \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\vec{\mathbf{i}}_{\text{U}}\right] \\ &= \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\left[\vec{\mathbf{i}}_{\text{E}} \quad \vec{\mathbf{i}}_{\text{N}} \quad \vec{\mathbf{i}}_{\text{U}}\right] \\ &= \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\end{aligned} \tag{B.142}$$

$$\mathbf{C}_{\text{ENU}}^{\text{RPY}} = \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}}); \tag{B.143}$$

that is, $\mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})$ is the coordinate transformation matrix from RPY coordinates to ENU coordinates. As a consequence, from Eq. B.226,

$$\begin{aligned}\frac{d}{dt}\mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}}) &= \frac{d}{dt}\mathbf{C}_{\text{ENU}}^{\text{RPY}} \\ &= \begin{bmatrix}0 & \omega_{\text{U}} & -\omega_{\text{N}}\\ -\omega_{\text{U}} & 0 & \omega_{\text{E}}\\ \omega_{\text{N}} & -\omega_{\text{E}} & 0\end{bmatrix}\mathbf{C}_{\text{ENU}}^{\text{RPY}}\end{aligned} \tag{B.144}$$

$$+\,\mathbf{C}_{\text{ENU}}^{\text{RPY}}\begin{bmatrix}0 & -\omega_{\text{Y}} & \omega_{\text{P}}\\ \omega_{\text{Y}} & 0 & -\omega_{\text{R}}\\ -\omega_{\text{P}} & \omega_{\text{R}} & 0\end{bmatrix}. \tag{B.145}$$

$$\frac{d}{dt}\mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}}) = \begin{bmatrix}0 & \omega_{\text{U}} & -\omega_{\text{N}}\\ -\omega_{\text{U}} & 0 & \omega_{\text{E}}\\ \omega_{\text{N}} & -\omega_{\text{E}} & 0\end{bmatrix}\mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}}) + \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})\begin{bmatrix}0 & -\omega_{\text{Y}} & \omega_{\text{P}}\\ \omega_{\text{Y}} & 0 & -\omega_{\text{R}}\\ -\omega_{\text{P}} & \omega_{\text{R}} & 0\end{bmatrix}, \tag{B.146}$$

where

$$\vec{\boldsymbol{\omega}}_{\text{RPY}} = \begin{bmatrix}\omega_{\text{R}}\\ \omega_{\text{P}}\\ \omega_{\text{Y}}\end{bmatrix} \tag{B.147}$$

is the vector of inertial rotation rates of the vehicle body, expressed in RPY coordinates, and

$$\vec{\boldsymbol{\omega}}_{\text{ENU}} = \begin{bmatrix}\omega_{\text{E}}\\ \omega_{\text{N}}\\ \omega_{\text{U}}\end{bmatrix} \tag{B.148}$$

is the vector of inertial rotation rates of the ENU coordinate frame, expressed in ENU coordinates.

The 3×3 matrix Eq. B.146 is equivalent to nine scalar equations:

$$\frac{\partial c_{11}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{11}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{11}}{\partial \rho_U}\dot{\rho}_U = -c_{1,3}\omega_P + c_{1,2}\omega_Y - c_{3,1}\omega_N + c_{2,1}\omega_U$$

$$\frac{\partial c_{12}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{12}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{12}}{\partial \rho_U}\dot{\rho}_U = c_{1,3}\omega_R - c_{1,1}\omega_Y - c_{3,2}\omega_N + c_{2,2}\omega_U$$

$$\frac{\partial c_{13}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{13}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{13}}{\partial \rho_U}\dot{\rho}_U = -c_{1,2}\omega_R + c_{1,1}\omega_P - c_{3,3}\omega_N + c_{2,3}\omega_U$$

$$\frac{\partial c_{21}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{21}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{21}}{\partial \rho_U}\dot{\rho}_U = -c_{2,3}\omega_P + c_{2,2}\omega_Y + c_{3,1}\omega_E - c_{1,1}\omega_U$$

$$\frac{\partial c_{22}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{22}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{22}}{\partial \rho_U}\dot{\rho}_U = c_{2,3}\omega_R - c_{2,1}\omega_Y + c_{3,2}\omega_E - c_{1,2}\omega_U$$

$$\frac{\partial c_{23}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{23}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{23}}{\partial \rho_U}\dot{\rho}_U = -c_{2,2}\omega_R + c_{2,1}\omega_P + c_{3,3}\omega_E - c_{1,3}\omega_U$$

$$\frac{\partial c_{31}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{31}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{31}}{\partial \rho_U}\dot{\rho}_U = -c_{3,3}\omega_P + c_{3,2}\omega_Y - c_{2,1}\omega_E + c_{1,1}\omega_N$$

$$\frac{\partial c_{32}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{32}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{32}}{\partial \rho_U}\dot{\rho}_U = c_{3,3}\omega_R - c_{3,1}\omega_Y - c_{2,2}\omega_E + c_{1,2}\omega_N$$

$$\frac{\partial c_{33}}{\partial \rho_E}\dot{\rho}_E + \frac{\partial c_{33}}{\partial \rho_N}\dot{\rho}_N + \frac{\partial c_{33}}{\partial \rho_U}\dot{\rho}_U = -c_{3,2}\omega_R + c_{3,1}\omega_P - c_{2,3}\omega_E + c_{1,3}\omega_N,$$

where

$$\begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \stackrel{\text{def}}{=} \mathbf{C}(\vec{\boldsymbol{\rho}}_{\text{ENU}})$$

and the partial derivatives

$$\frac{\partial c_{11}}{\partial \rho_E} = \frac{u_E\left(1-u_E^2\right)[2\,(1-\cos(\theta))-\theta\sin(\theta)]}{\theta}$$

$$\frac{\partial c_{11}}{\partial \rho_N} = \frac{u_N\left[-2u_E^2\,(1-\cos(\theta))-\theta\sin(\theta)\left(1-u_E^2\right)\right]}{\theta}$$

$$\frac{\partial c_{11}}{\partial \rho_U} = \frac{u_U\left[-2u_E^2\,(1-\cos(\theta))-\theta\sin(\theta)\left(1-u_E^2\right)\right]}{\theta}$$

$$\frac{\partial c_{12}}{\partial \rho_E} = \frac{u_N\left(1-2u_E^2\right)(1-\cos(\theta))+u_E u_U\sin(\theta)-\theta u_E u_U\cos(\theta)+\theta u_N u_E^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{12}}{\partial \rho_{\mathrm{N}}} = \frac{u_{\mathrm{E}}\left(1-2u_{\mathrm{N}}^2\right)(1-\cos(\theta)) + u_{\mathrm{U}}u_{\mathrm{N}}\sin(\theta) - \theta u_{\mathrm{N}}u_{\mathrm{U}}\cos(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{N}}^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{12}}{\partial \rho_{\mathrm{U}}} = \frac{-2u_{\mathrm{E}}u_{\mathrm{N}}u_{\mathrm{U}}(1-\cos(\theta)) - \left(1-u_{\mathrm{U}}^2\right)\sin(\theta) - \theta u_{\mathrm{U}}^2\cos(\theta) + \theta u_{\mathrm{U}}u_{\mathrm{N}}u_{\mathrm{E}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{13}}{\partial \rho_{\mathrm{E}}} = \frac{u_{\mathrm{U}}\left(1-2u_{\mathrm{E}}^2\right)(1-\cos(\theta)) - u_{\mathrm{E}}u_{\mathrm{N}}\sin(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{N}}\cos(\theta) + \theta u_{\mathrm{U}}u_{\mathrm{E}}^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{13}}{\partial \rho_{\mathrm{N}}} = \frac{-2u_{\mathrm{E}}u_{\mathrm{N}}u_{\mathrm{U}}(1-\cos(\theta)) + \left(1-u_{\mathrm{N}}^2\right)\sin(\theta) + \theta u_{\mathrm{N}}^2\cos(\theta) + \theta u_{\mathrm{U}}u_{\mathrm{N}}u_{\mathrm{E}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{13}}{\partial \rho_{\mathrm{U}}} = \frac{u_{\mathrm{E}}\left(1-2u_{\mathrm{U}}^2\right)(1-\cos(\theta)) - u_{\mathrm{U}}u_{\mathrm{N}}\sin(\theta) + \theta u_{\mathrm{N}}u_{\mathrm{U}}\cos(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{U}}^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{21}}{\partial \rho_{\mathrm{E}}} = \frac{u_{\mathrm{N}}\left(1-2u_{\mathrm{E}}^2\right)(1-\cos(\theta)) - u_{\mathrm{E}}u_{\mathrm{U}}\sin(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{U}}\cos(\theta) + \theta u_{\mathrm{N}}u_{\mathrm{E}}^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{21}}{\partial \rho_{\mathrm{N}}} = \frac{u_{\mathrm{E}}\left(1-2u_{\mathrm{N}}^2\right)(1-\cos(\theta)) - u_{\mathrm{U}}u_{\mathrm{N}}\sin(\theta) + \theta u_{\mathrm{N}}u_{\mathrm{U}}\cos(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{N}}^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{21}}{\partial \rho_{\mathrm{U}}} = \frac{-2u_{\mathrm{E}}u_{\mathrm{N}}u_{\mathrm{U}}(1-\cos(\theta)) + \sin(\theta)\left(1-u_{\mathrm{U}}^2\right) + \theta u_{\mathrm{U}}^2\cos(\theta) + \theta u_{\mathrm{U}}u_{\mathrm{N}}u_{\mathrm{E}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{22}}{\partial \rho_{\mathrm{E}}} = \frac{u_{\mathrm{E}}\left[-2u_{\mathrm{N}}^2(1-\cos(\theta)) - \theta\left(1-u_{\mathrm{N}}^2\right)\sin(\theta)\right]}{\theta}$$

$$\frac{\partial c_{22}}{\partial \rho_{\mathrm{N}}} = \frac{u_{\mathrm{N}}\left[1-u_{\mathrm{N}}^2\right]\left[2(1-\cos(\theta)) - \theta\sin(\theta)\right]}{\theta}$$

$$\frac{\partial c_{22}}{\partial \rho_{\mathrm{U}}} = \frac{u_{\mathrm{U}}\left[-2u_{\mathrm{N}}^2(1-\cos(\theta)) - \theta\left(1-u_{\mathrm{N}}^2\right)\sin(\theta)\right]}{\theta}$$

$$\frac{\partial c_{23}}{\partial \rho_{\mathrm{E}}} = \frac{-2u_{\mathrm{E}}u_{\mathrm{N}}u_{\mathrm{U}}(1-\cos(\theta)) - \left(1-u_{\mathrm{E}}^2\right)\sin(\theta) - \theta u_{\mathrm{E}}^2\cos(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{N}}u_{\mathrm{U}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{23}}{\partial \rho_{\mathrm{N}}} = \frac{u_{\mathrm{U}}\left(1-2u_{\mathrm{N}}^2\right)(1-\cos(\theta)) + u_{\mathrm{E}}u_{\mathrm{N}}\sin(\theta) - \theta u_{\mathrm{E}}u_{\mathrm{N}}\cos(\theta) + \theta u_{\mathrm{N}}^2u_{\mathrm{U}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{23}}{\partial \rho_{\mathrm{U}}} = \frac{u_{\mathrm{N}}\left(1-2u_{\mathrm{U}}^2\right)(1-\cos(\theta)) + u_{\mathrm{E}}u_{\mathrm{U}}\sin(\theta) - \theta u_{\mathrm{E}}u_{\mathrm{U}}\cos(\theta) + \theta u_{\mathrm{U}}^2u_{\mathrm{N}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{31}}{\partial \rho_{\mathrm{E}}} = \frac{u_{\mathrm{U}}\left(1-2u_{\mathrm{E}}^2\right)(1-\cos(\theta)) + u_{\mathrm{E}}u_{\mathrm{N}}\sin(\theta) - \theta u_{\mathrm{E}}u_{\mathrm{N}}\cos(\theta) + \theta u_{\mathrm{U}}u_{\mathrm{E}}^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{31}}{\partial \rho_{\mathrm{N}}} = \frac{-2u_{\mathrm{E}}u_{\mathrm{N}}u_{\mathrm{U}}(1-\cos(\theta)) - \left(1-u_{\mathrm{N}}^2\right)\sin(\theta) - \theta u_{\mathrm{N}}^2\cos(\theta) + \theta u_{\mathrm{U}}u_{\mathrm{N}}u_{\mathrm{E}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{31}}{\partial \rho_{\mathrm{U}}} = \frac{u_{\mathrm{E}}\left(1-2u_{\mathrm{U}}^2\right)(1-\cos(\theta)) + u_{\mathrm{U}}u_{\mathrm{N}}\sin(\theta) - \theta u_{\mathrm{N}}u_{\mathrm{U}}\cos(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{U}}^2\sin(\theta)}{\theta}$$

$$\frac{\partial c_{32}}{\partial \rho_{\mathrm{E}}} = \frac{-2u_{\mathrm{E}}u_{\mathrm{N}}u_{\mathrm{U}}(1-\cos(\theta)) + \left(1-u_{\mathrm{E}}^2\right)\sin(\theta) + \theta u_{\mathrm{E}}^2\cos(\theta) + \theta u_{\mathrm{U}}u_{\mathrm{N}}u_{\mathrm{E}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{32}}{\partial \rho_{\mathrm{N}}} = \frac{u_{\mathrm{U}}\left(1-2u_{\mathrm{N}}^2\right)(1-\cos(\theta)) - u_{\mathrm{E}}u_{\mathrm{N}}\sin(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{N}}\cos(\theta) + \theta u_{\mathrm{N}}^2u_{\mathrm{U}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{32}}{\partial \rho_{\mathrm{U}}} = \frac{u_{\mathrm{N}}\left(1-2u_{\mathrm{U}}^2\right)(1-\cos(\theta)) - u_{\mathrm{E}}u_{\mathrm{U}}\sin(\theta) + \theta u_{\mathrm{E}}u_{\mathrm{U}}\cos(\theta) + \theta u_{\mathrm{U}}^2u_{\mathrm{N}}\sin(\theta)}{\theta}$$

$$\frac{\partial c_{33}}{\partial \rho_{\mathrm{E}}} = \frac{u_{\mathrm{E}}\left[-2u_{\mathrm{U}}^2(1-\cos(\theta)) - \theta\sin(\theta)\left(1+u_{\mathrm{U}}^2\right)\right]}{\theta}$$

$$\frac{\partial c_{33}}{\partial \rho_{\mathrm{N}}} = \frac{u_{\mathrm{N}}\left[-2u_{\mathrm{U}}^2(1-\cos(\theta)) - \theta\sin(\theta)\left(1+u_{\mathrm{U}}^2\right)\right]}{\theta}$$

$$\frac{\partial c_{33}}{\partial \rho_{\mathrm{U}}} = \frac{u_{\mathrm{U}}\left[1-u_{\mathrm{U}}^2\right]\left[2(1-\cos(\theta)) - \theta\sin(\theta)\right]}{\theta}$$

for

$$\theta \stackrel{\text{def}}{=} |\vec{\rho}_{\mathrm{ENU}}|$$
$$u_{\mathrm{E}} \stackrel{\text{def}}{=} \rho_{\mathrm{E}}/\theta$$
$$u_{\mathrm{N}} \stackrel{\text{def}}{=} \rho_{\mathrm{N}}/\theta$$
$$u_{\mathrm{U}} \stackrel{\text{def}}{=} \rho_{\mathrm{U}}/\theta.$$

These nine scalar linear equations can be put into matrix form and solved in least-squares fashion as

$$\mathbf{L}\begin{bmatrix}\dot{\rho}_{\mathrm{E}}\\ \dot{\rho}_{\mathrm{N}}\\ \dot{\rho}_{\mathrm{U}}\end{bmatrix} = \mathbf{R}\begin{bmatrix}\omega_{\mathrm{R}}\\ \omega_{\mathrm{P}}\\ \omega_{\mathrm{Y}}\\ \omega_{\mathrm{E}}\\ \omega_{\mathrm{N}}\\ \omega_{\mathrm{U}}\end{bmatrix} \tag{B.149}$$

$$\begin{bmatrix}\dot{\rho}_{\mathrm{E}}\\ \dot{\rho}_{\mathrm{N}}\\ \dot{\rho}_{\mathrm{U}}\end{bmatrix} = \underbrace{\left[\mathbf{L}^T\mathbf{L}\right]\backslash\left[\mathbf{L}^T\mathbf{R}\right]}_{\frac{\partial\dot{\rho}}{\partial\vec{\omega}}}\begin{bmatrix}\vec{\omega}_{\mathrm{RPY}}\\ \vec{\omega}_{\mathrm{ENU}}\end{bmatrix}. \tag{B.150}$$

The matrix product $\mathbf{L}^{\mathrm{T}}\mathbf{L}$ will always be invertible because its determinant

$$\det\left[\mathbf{L}^{\mathrm{T}}\mathbf{L}\right] = 32\frac{(1-\cos(\theta))^2}{\theta^4} \tag{B.151}$$

$$\lim_{\theta\to 0}\det\left[\mathbf{L}^{\mathrm{T}}\mathbf{L}\right] = 8, \tag{B.152}$$

and the resulting equation for $\dot{\boldsymbol{\rho}}_{\mathrm{ENU}}$ can be put into the form

$$\dot{\boldsymbol{\rho}}_{ENU} = \left[\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}}\right]\begin{bmatrix}\vec{\boldsymbol{\omega}}_{\mathrm{RPY}}\\ \vec{\boldsymbol{\omega}}_{\mathrm{ENU}}\end{bmatrix}. \tag{B.153}$$

The 3×6 matrix $\partial\dot{\boldsymbol{\rho}}/\partial\vec{\boldsymbol{\omega}}$ can be partitioned as

$$\left[\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}}\right] = \left[\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}_{\mathrm{RPY}}}\middle|\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}_{\mathrm{ENU}}}\right] \tag{B.154}$$

with 3×3 submatrices

$$\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}_{\mathrm{RPY}}} = \left[\frac{1}{|\vec{\boldsymbol{\rho}}|^2} - \frac{\sin(|\vec{\boldsymbol{\rho}}|)}{2|\vec{\boldsymbol{\rho}}|[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\right]\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^{\mathrm{T}} + \frac{|\vec{\boldsymbol{\rho}}|\sin(|\vec{\boldsymbol{\rho}}|)}{2[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\mathbf{I} + \frac{1}{2}[\vec{\boldsymbol{\rho}}\otimes] \tag{B.155}$$

$$= \begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}^{\mathrm{T}} + \frac{\theta\sin(\theta)}{2[1-\cos(\theta)]}\left[\mathbf{I} - \begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}^{T}\right] + \frac{\theta}{2}\left[\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}\otimes\right] \tag{B.156}$$

$$\lim_{|\vec{\boldsymbol{\rho}}|\to 0}\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}_{\mathrm{RPY}}} = \mathbf{I} \tag{B.157}$$

$$\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}_{\mathrm{ENU}}} = -\left[\frac{1}{|\vec{\boldsymbol{\rho}}|^2} - \frac{\sin(|\vec{\boldsymbol{\rho}}|)}{2|\vec{\boldsymbol{\rho}}|[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\right]\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^{\mathrm{T}} - \frac{|\vec{\boldsymbol{\rho}}|\sin(|\vec{\boldsymbol{\rho}}|)}{2[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\mathbf{I} + \frac{1}{2}[\vec{\boldsymbol{\rho}}\otimes] \tag{B.158}$$

$$= -\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}^{\mathrm{T}} - \frac{\theta\sin(\theta)}{2[1-\cos(\theta)]}\left[\mathbf{I} - \begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}^{\mathrm{T}}\right] + \frac{\theta}{2}\left[\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}\otimes\right] \tag{B.159}$$

$$\lim_{|\vec{\boldsymbol{\rho}}|\to 0}\frac{\partial\dot{\boldsymbol{\rho}}}{\partial\vec{\boldsymbol{\omega}}_{\mathrm{ENU}}} = -\mathbf{I}. \tag{B.160}$$

For locally leveled gimbaled systems, $\vec{\boldsymbol{\omega}}_{\mathrm{RPY}} = \vec{\mathbf{0}}$; that is, the gimbals normally keep the accelerometer axes aligned to the ENU or NED coordinate axes, a process which is modeled by $\vec{\boldsymbol{\omega}}_{\mathrm{ENU}}$ alone.

B.4.2.5 Time Derivatives of Matrix Expressions The Kalman filter implementation for integrating GNSS with a strapdown INS in Chapter 8 will require derivatives with respect to time of the matrices

$$\frac{\partial \dot{\boldsymbol{\rho}}_{\text{ENU}}}{\partial \vec{\boldsymbol{\omega}}_{\text{RPH}}} \text{ (Eq. B.155) and } \frac{\partial \dot{\boldsymbol{\rho}}_{\text{ENU}}}{\partial \vec{\boldsymbol{\omega}}_{\text{ENU}}} \text{ (Eq. B.158).}$$

We derive here a general-purpose formula for taking such derivatives and then apply it to these two cases.

General Formulas There is a general-purpose formula for taking the time derivatives

$$\frac{d}{dt}\mathbf{M}(\vec{\boldsymbol{\rho}})$$

of matrix expressions of the sort

$$\mathbf{M}(\vec{\boldsymbol{\rho}}) = \mathbf{M}(s_1(\vec{\boldsymbol{\rho}}), s_2(\vec{\boldsymbol{\rho}}), s_3(\vec{\boldsymbol{\rho}})) \tag{B.161}$$

$$= s_1(\vec{\boldsymbol{\rho}})\mathbf{I}_3 + s_2(\vec{\boldsymbol{\rho}})[\vec{\boldsymbol{\rho}}\otimes] + s_3(\vec{\boldsymbol{\rho}})\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^{\mathrm{T}}; \tag{B.162}$$

that is, as linear combinations of $\mathbf{I}_3$, $\vec{\boldsymbol{\rho}}\otimes$, and $\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^T$ with scalar functions of $\vec{\boldsymbol{\rho}}$ as the coefficients.

The derivation uses the time derivatives of the basis matrices,

$$\frac{d}{dt}\mathbf{I}_3 = \mathbf{0}_3 \tag{B.163}$$

$$\frac{d}{dt}[\vec{\boldsymbol{\rho}}\otimes] = [\dot{\boldsymbol{\rho}}\otimes] \tag{B.164}$$

$$\frac{d}{dt}\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^T = \dot{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^T + \vec{\boldsymbol{\rho}}\dot{\boldsymbol{\rho}}^T, \tag{B.165}$$

where the vector

$$\dot{\boldsymbol{\rho}} = \frac{d}{dt}\vec{\boldsymbol{\rho}}, \tag{B.166}$$

and then uses the chain rule for differentiation to obtain the general formula:

$$\begin{aligned}\frac{d}{dt}\mathbf{M}(\vec{\boldsymbol{\rho}}) = {} & \frac{\partial s_1(\vec{\boldsymbol{\rho}})}{\partial \vec{\boldsymbol{\rho}}}\dot{\boldsymbol{\rho}}\mathbf{I}_3 + \frac{\partial s_2(\vec{\boldsymbol{\rho}})}{\partial \vec{\boldsymbol{\rho}}}\dot{\boldsymbol{\rho}}[\vec{\boldsymbol{\rho}}\otimes] + s_2(\vec{\boldsymbol{\rho}})[\dot{\boldsymbol{\rho}}\otimes] \\ & + \frac{\partial s_3(\vec{\boldsymbol{\rho}})}{\partial \vec{\boldsymbol{\rho}}}\dot{\boldsymbol{\rho}}[\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^T] + s_3(\vec{\boldsymbol{\rho}})[\dot{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^T + \vec{\boldsymbol{\rho}}\dot{\boldsymbol{\rho}}^T],\end{aligned} \tag{B.167}$$

where the gradients $\partial s_i(\partial\vec{\boldsymbol{\rho}})/\partial\vec{\boldsymbol{\rho}}$ are to be computed as row vectors and the inner products

$$\frac{\partial s_i(\vec{\boldsymbol{\rho}})}{\partial \vec{\boldsymbol{\rho}}}\dot{\boldsymbol{\rho}}$$

will be scalars.

Equation B.167 is the general-purpose formula for the matrix forms of interest, which differ only in their scalar functions $s_i(\vec{\rho})$. The scalar functions $s_i(\vec{\rho})$ are generally rational functions of the following scalar functions (shown in terms of their gradients):

$$\frac{\partial}{\partial\vec{\rho}}|\vec{\rho}|^p = p|\vec{\rho}|^{p-2}\vec{\rho}^{\mathrm{T}} \tag{B.168}$$

$$\frac{\partial}{\partial\vec{\rho}}\sin(|\vec{\rho}|) = \cos(|\vec{\rho}|)|\vec{\rho}|^{-1}\vec{\rho}^{\mathrm{T}} \tag{B.169}$$

$$\frac{\partial}{\partial\vec{\rho}}\cos(|\vec{\rho}|) = -\sin(|\vec{\rho}|)|\vec{\rho}|^{-1}\vec{\rho}^{\mathrm{T}}. \tag{B.170}$$

Time Derivative of $\partial\dot{\vec{\rho}}_{\mathrm{ENU}}/\partial\vec{\omega}_{\mathrm{RPY}}$ In this case (Eq. B.155),

$$s_1(\vec{\rho}) = \frac{|\vec{\rho}|\sin(|\vec{\rho}|)}{2[1-\cos(|\vec{\rho}|)]} \tag{B.171}$$

$$\frac{\partial s_1(\vec{\rho})}{\partial\vec{\rho}} = -\frac{1-|\vec{\rho}|^{-1}\sin(|\vec{\rho}|)}{2[1-\cos(|\vec{\rho}|)]}\vec{\rho}^{\mathrm{T}} \tag{B.172}$$

$$s_2(\vec{\rho}) = \frac{1}{2} \tag{B.173}$$

$$\frac{\partial s_2}{\partial\vec{\rho}} = 0_{1\times 3} \tag{B.174}$$

$$s_3(\vec{\rho}) = \left[\frac{1}{|\vec{\rho}|^2} - \frac{\sin(|\vec{\rho}|)}{2|\vec{\rho}|[1-\cos(|\vec{\rho}|)]}\right] \tag{B.175}$$

$$\frac{\partial s_3(\vec{\rho})}{\partial\vec{\rho}} = \frac{1+|\vec{\rho}|^{-1}\sin(|\vec{\rho}|)-4|\vec{\rho}|^{-2}[1-\cos(|\vec{\rho}|)]}{2|\vec{\rho}|^2[1-\cos(|\vec{\rho}|)]}\vec{\rho}^{\mathrm{T}} \tag{B.176}$$

$$\begin{aligned}\frac{d}{dt}\frac{\partial\dot{\vec{\rho}}_{\mathrm{ENU}}}{\partial\vec{\omega}_{\mathrm{RPY}}} &= \frac{\partial s_1(\vec{\rho})}{\partial\vec{\rho}}\dot{\rho}\mathbf{I}_3 + \frac{\partial s_2(\vec{\rho})}{\partial\vec{\rho}}\dot{\rho}[\vec{\rho}\otimes] + s_2(\vec{\rho})[\dot{\rho}\otimes] \\ &\quad + \frac{\partial s_3(\vec{\rho})}{\partial\vec{\rho}}\dot{\rho}[\vec{\rho}\vec{\rho}^{\mathrm{T}}] + s_3(\vec{\rho})[\dot{\rho}\vec{\rho}^{\mathrm{T}} + \vec{\rho}\dot{\rho}^{\mathrm{T}}] \\ &= -\left\{\frac{1-|\vec{\rho}|^{-1}\sin(|\vec{\rho}|)}{2[1-\cos(|\vec{\rho}|)]}\vec{\rho}^{\mathrm{T}}\right\}(\vec{\rho}^{\mathrm{T}}\dot{\rho})\mathbf{I}_3 + \frac{1}{2}(\vec{\rho})[\dot{\rho}\otimes] \\ &\quad + \left\{\frac{1+|\vec{\rho}|^{-1}\sin(|\vec{\rho}|)-4|\vec{\rho}|^{-2}[1-\cos(|\vec{\rho}|)]}{2|\vec{\rho}|^2[1-\cos(|\vec{\rho}|)]}\right\}\times(\vec{\rho}^{\mathrm{T}}\dot{\rho})[\vec{\rho}\vec{\rho}^{\mathrm{T}}]\end{aligned} \tag{B.177}$$

$$+\left[\frac{1}{|\vec{\rho}|^2} - \frac{\sin(|\vec{\rho}|)}{2|\vec{\rho}|[1-\cos(|\vec{\rho}|)]}\right][\dot{\rho}\vec{\rho}^{\mathrm{T}} + \vec{\rho}\dot{\rho}^{\mathrm{T}}]. \tag{B.178}$$

Time Derivative of $\partial\dot{\vec{\boldsymbol{\rho}}}_{\text{ENU}}/\partial\vec{\boldsymbol{\omega}}_{\text{ENU}}$ In this case (Eq. B.158),

$$s_1(\vec{\boldsymbol{\rho}}) = -\frac{|\vec{\boldsymbol{\rho}}|\sin(|\vec{\boldsymbol{\rho}}|)}{2[1-\cos(|\vec{\boldsymbol{\rho}}|)]} \tag{B.179}$$

$$\frac{\partial s_1(\vec{\boldsymbol{\rho}})}{\partial\vec{\boldsymbol{\rho}}} = \frac{1-|\vec{\boldsymbol{\rho}}|^{-1}\sin(|\vec{\boldsymbol{\rho}}|)}{2[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\vec{\boldsymbol{\rho}}^{\mathrm{T}} \tag{B.180}$$

$$s_2(\vec{\boldsymbol{\rho}}) = \frac{1}{2} \tag{B.181}$$

$$\frac{\partial s_2}{\partial\vec{\boldsymbol{\rho}}} = 0_{1\times 3} \tag{B.182}$$

$$s_3(\vec{\boldsymbol{\rho}}) = -\left[\frac{1}{|\vec{\boldsymbol{\rho}}|^2} - \frac{\sin(|\vec{\boldsymbol{\rho}}|)}{2|\vec{\boldsymbol{\rho}}|[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\right] \tag{B.183}$$

$$\frac{\partial s_3(\vec{\boldsymbol{\rho}})}{\partial\vec{\boldsymbol{\rho}}} = -\frac{1+|\vec{\boldsymbol{\rho}}|^{-1}\sin(|\vec{\boldsymbol{\rho}}|)-4|\vec{\boldsymbol{\rho}}|^{-2}[1-\cos(|\vec{\boldsymbol{\rho}}|)]}{2|\vec{\boldsymbol{\rho}}|^2[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\vec{\boldsymbol{\rho}}^{\mathrm{T}} \tag{B.184}$$

$$\begin{aligned}\frac{d}{dt}\frac{\partial\dot{\vec{\boldsymbol{\rho}}}_{\text{ENU}}}{\partial\vec{\boldsymbol{\omega}}_{\text{ENU}}} &= \frac{\partial s_1(\vec{\boldsymbol{\rho}})}{\partial\vec{\boldsymbol{\rho}}}\dot{\boldsymbol{\rho}}\mathbf{I}_3 + \frac{\partial s_2(\vec{\boldsymbol{\rho}})}{\partial\vec{\boldsymbol{\rho}}}\dot{\boldsymbol{\rho}}[\vec{\boldsymbol{\rho}}\otimes] + s_2(\vec{\boldsymbol{\rho}})[\dot{\boldsymbol{\rho}}\otimes] \\ &\quad + \frac{\partial s_3(\vec{\boldsymbol{\rho}})}{\partial\vec{\boldsymbol{\rho}}}\dot{\boldsymbol{\rho}}[\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^{\mathrm{T}}] + s_3(\vec{\boldsymbol{\rho}})[\dot{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^{\mathrm{T}} + \vec{\boldsymbol{\rho}}\dot{\boldsymbol{\rho}}^{\mathrm{T}}] \\ &= \left\{\frac{1-|\vec{\boldsymbol{\rho}}|^{-1}\sin(|\vec{\boldsymbol{\rho}}|)}{2[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\vec{\boldsymbol{\rho}}^{\mathrm{T}}\right\}(\vec{\boldsymbol{\rho}}^{\mathrm{T}}\dot{\boldsymbol{\rho}})\mathbf{I}_3 + \frac{1}{2}(\vec{\boldsymbol{\rho}})[\dot{\boldsymbol{\rho}}\otimes] \\ &\quad - \left\{\frac{1+|\vec{\boldsymbol{\rho}}|^{-1}\sin(|\vec{\boldsymbol{\rho}}|)-4|\vec{\boldsymbol{\rho}}|^{-2}[1-\cos(|\vec{\boldsymbol{\rho}}|)]}{2|\vec{\boldsymbol{\rho}}|^2[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\vec{\boldsymbol{\rho}}^{\mathrm{T}}\right\} \\ &\quad \times(\vec{\boldsymbol{\rho}}^{\mathrm{T}}\dot{\boldsymbol{\rho}})[\vec{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^{\mathrm{T}}]\end{aligned} \tag{B.185}$$

$$-\left[\frac{1}{|\vec{\boldsymbol{\rho}}|^2} - \frac{\sin(|\vec{\boldsymbol{\rho}}|)}{2|\vec{\boldsymbol{\rho}}|[1-\cos(|\vec{\boldsymbol{\rho}}|)]}\right](\vec{\boldsymbol{\rho}})[\dot{\boldsymbol{\rho}}\vec{\boldsymbol{\rho}}^{\mathrm{T}} + \vec{\boldsymbol{\rho}}\dot{\boldsymbol{\rho}}^{\mathrm{T}}]. \tag{B.186}$$

B.4.2.6 Partial Derivatives with Respect to Rotation Vectors Calculation of the dynamic coefficient matrices $\mathbf{F}$ and measurement sensitivity matrices $\mathbf{H}$ in linearized or extended Kalman filtering with rotation vectors $\vec{\boldsymbol{\rho}}_{\text{ENU}}$ as part of the system model state vector requires taking derivatives with respect to $\vec{\boldsymbol{\rho}}_{\text{ENU}}$ of associated vector-valued **f**- or **h**-functions, as

$$\mathbf{F} = \frac{\partial\mathbf{f}(\vec{\boldsymbol{\rho}}_{\text{ENU}}, \mathbf{v})}{\partial\vec{\boldsymbol{\rho}}_{\text{ENU}}} \tag{B.187}$$

$$\mathbf{H} = \frac{\partial\mathbf{h}(\vec{\boldsymbol{\rho}}_{\text{ENU}}, \mathbf{v})}{\partial\vec{\boldsymbol{\rho}}_{\text{ENU}}}, \tag{B.188}$$

where the vector-valued functions will have the general form

$$\begin{aligned}&\mathbf{f}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}, \mathbf{v}) \text{ or } \mathbf{h}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}, \mathbf{v})\\ &\quad = \left\{ s_0(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})\mathbf{I}_3 + s_1(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})[\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}\otimes] + s_2(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{T} \right\}\mathbf{v},\end{aligned} \tag{B.189}$$

and

s_0, s_1, s_2 are scalar-valued functions of $\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}$, and
$\mathbf{v}$ is a vector that does not depend on $\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}$.

We will derive here the general formulas that can be used for taking the partial derivatives

$$\frac{\partial \mathbf{f}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}, \mathbf{v})}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} \text{ or } \frac{\partial \mathrm{h}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}, \mathbf{v})}{\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}}.$$

These formulas can all be derived by calculating the derivatives of the different factors in the functional forms and then using the chain rule for differentiation to obtain the final result.

Derivatives of Scalars The derivatives of the scalar factors s_0, s_1, s_2 will be

$$\frac{\partial}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) = \left[\frac{\partial s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{E}}} \quad \frac{\partial s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{N}}} \quad \frac{\partial s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{U}}} \right], \tag{B.190}$$

a row vector. Consequently, for any vector-valued function $\mathbf{g}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})$, by the chain rule, the derivatives of the vector-valued product $s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})\ \mathbf{g}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})$ will be

$$\frac{\partial \{ s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})\, g(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) \}}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} = \underbrace{\mathrm{g}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) \frac{\partial s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}}}_{3\times 3 \text{ matrix}} + s_i(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) \underbrace{\frac{\partial \mathbf{g}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}}}_{3\times 3 \text{ matrix}}, \tag{B.191}$$

the result of which will be the 3×3 Jacobian matrix of that subexpression in $\mathbf{f}$ or $\mathbf{h}$.

Derivatives of Vectors The three potential forms of the vector-valued function $\mathbf{g}$ in Eq. B.191 are

$$\mathbf{g}(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) = \begin{cases} \mathbf{I}\mathbf{v} = \mathbf{v} \\ \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \otimes \mathbf{v} \\ \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \mathbf{v}, \end{cases} \tag{B.192}$$

each of which is considered independently:

$$\frac{\partial \mathbf{v}}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} = 0_{3\times 3} \tag{B.193}$$

$$\frac{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \otimes \mathbf{v}}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} = \frac{\partial [-\mathbf{v} \otimes \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}]}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} \tag{B.194}$$

$$= -[\mathbf{v}\otimes] \tag{B.195}$$

$$= -\begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \tag{B.196}$$

$$\frac{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \mathbf{v}}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} = \left(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \mathbf{v}\right) \frac{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} + \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \frac{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \mathbf{v}}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} \tag{B.197}$$

$$= \left(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \mathbf{v}\right) \mathbf{I}_{3\times 3} + \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \mathbf{v}^{\mathrm{T}}. \tag{B.198}$$

General Formula Combining the above formulas for the different parts, one can obtain the following general-purpose formula:

$$\begin{aligned} &\frac{\partial}{\partial \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}} \left\{ s_0(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) \mathbf{I}_3 + s_1(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) [\vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \otimes] + s_2(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \right\} \mathbf{v} \\ &= \mathbf{v} \begin{bmatrix} \dfrac{\partial s_0(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{E}}} & \dfrac{\partial s_0(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{N}}} & \dfrac{\partial s_0(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{U}}} \end{bmatrix} \\ &+ [\vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \otimes \mathbf{v}] \begin{bmatrix} \dfrac{\partial s_1(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{E}}} & \dfrac{\partial s_1(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{N}}} & \dfrac{\partial s_1(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{U}}} \end{bmatrix} - s_1(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) [\mathbf{v}\otimes] \\ &+ \left(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \mathbf{v}\right) \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \begin{bmatrix} \dfrac{\partial s_2(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{E}}} & \dfrac{\partial s_2(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{N}}} & \dfrac{\partial s_2(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}})}{\partial \rho_{\mathrm{U}}} \end{bmatrix} \\ &+ s_2(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}) \left[\left(\vec{\boldsymbol{\rho}}_{\mathrm{ENU}}^{\mathrm{T}} \mathbf{v}\right) \mathbf{I}_{3\times 3} + \vec{\boldsymbol{\rho}}_{\mathrm{ENU}} \mathbf{v}^{T} \right], \end{aligned} \tag{B.199}$$

applicable for any differentiable scalar functions s_0, s_1, s_2.

B.4.3 Direction Cosine Matrix

We have demonstrated in Eq. B.13 that the coordinate transformation matrix between one orthogonal coordinate system and another is a matrix of direction cosines between the unit axis vectors of the two coordinate systems,

$$\mathbf{C}_{\mathrm{XYZ}}^{\mathrm{UVW}} = \begin{bmatrix} \cos(\theta_{\mathrm{XU}}) & \cos(\theta_{\mathrm{XV}}) & \cos(\theta_{\mathrm{XW}}) \\ \cos(\theta_{\mathrm{YU}}) & \cos(\theta_{\mathrm{YV}}) & \cos(\theta_{\mathrm{YW}}) \\ \cos(\theta_{\mathrm{ZU}}) & \cos(\theta_{\mathrm{ZV}}) & \cos(\theta_{\mathrm{ZW}}) \end{bmatrix}. \tag{B.200}$$

Because the angles do not depend on the order of the direction vectors (i.e., $\theta_{ab} = \theta_{ba}$), the inverse transformation matrix

$$\mathbf{C}_{\mathrm{UVW}}^{\mathrm{XYZ}} = \begin{bmatrix} \cos(\theta_{\mathrm{UX}}) & \cos(\theta_{\mathrm{UY}}) & \cos(\theta_{\mathrm{UZ}}) \\ \cos(\theta_{\mathrm{VX}}) & \cos(\theta_{\mathrm{VY}}) & \cos(\theta_{\mathrm{VZ}}) \\ \cos(\theta_{\mathrm{WX}}) & \cos(\theta_{\mathrm{WY}}) & \cos(\theta_{\mathrm{WX}}) \end{bmatrix} \tag{B.201}$$

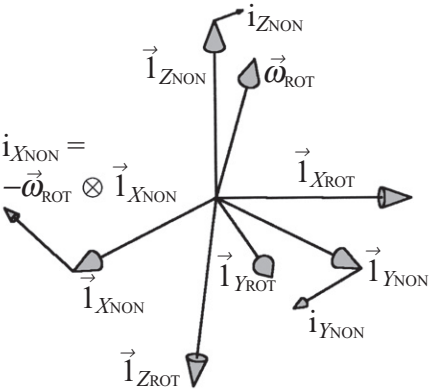


Fig. B.14 Rotating coordinates.

$$= \begin{bmatrix} \cos(\theta_{\mathrm{XU}}) & \cos(\theta_{\mathrm{XV}}) & \cos(\theta_{\mathrm{XW}}) \\ \cos(\theta_{\mathrm{YU}}) & \cos(\theta_{\mathrm{YV}}) & \cos(\theta_{\mathrm{YW}}) \\ \cos(\theta_{\mathrm{ZU}}) & \cos(\theta_{\mathrm{ZV}}) & \cos(\theta_{\mathrm{ZW}}) \end{bmatrix}^{\mathrm{T}} \tag{B.202}$$

$$= \left(\mathbf{C}_{\mathrm{XYZ}}^{\mathrm{UVW}}\right)^{\mathrm{T}}; \tag{B.203}$$

that is, the inverse coordinate transformation matrix is the transpose of the forward coordinate transformation matrix. This implies that the coordinate transformation matrices are orthogonal matrices.

B.4.3.1 Rotating Coordinates Let ROT denote a set of rotating coordinates, with axes X_{ROT}, Y_{ROT}, Z_{ROT}, and let NON represent a set of nonrotating (i.e., inertial) coordinates, with axes X_{NON}, Y_{NON}, Z_{NON}, as illustrated in Fig. B.14.

Any vector $\mathbf{v}_{\mathrm{ROT}}$ in rotating coordinates can be represented in terms of its nonrotating components and unit vectors parallel to the nonrotating axes, as

$$\mathbf{v}_{\mathrm{ROT}} = v_{x\mathrm{NON}}\vec{\mathbf{1}}_{x\mathrm{NON}} + v_{y\mathrm{NON}}\vec{\mathbf{1}}_{y\mathrm{NON}} + v_{z\mathrm{NON}}\vec{\mathbf{1}}_{z\mathrm{NON}} \tag{B.204}$$

$$= \begin{bmatrix} \vec{\mathbf{1}}_{x\mathrm{NON}} & \vec{\mathbf{1}}_{y\mathrm{NON}} & \vec{\mathbf{1}}_{z\mathrm{NON}} \end{bmatrix} \begin{bmatrix} v_{x\mathrm{NON}} \\ v_{y\mathrm{NON}} \\ v_{z\mathrm{NON}} \end{bmatrix} \tag{B.205}$$

$$= \mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}\mathbf{v}_{\mathrm{NON}}, \tag{B.206}$$

where

$v_{x\mathrm{NON}}$, $v_{y\mathrm{NON}}$, $v_{z\mathrm{NON}}$, are the nonrotating components of the vector;

$\vec{\mathbf{1}}_{x\mathrm{NON}}$, $\vec{\mathbf{1}}_{y\mathrm{NON}}$, $\vec{\mathbf{1}}_{z\mathrm{NON}}$ are unit vectors along the X_{NON}, Y_{NON}, Z_{NON} axes, as expressed in rotating coordinates;

$\mathbf{v}_{\mathrm{ROT}}$ is the vector $\mathbf{v}$ expressed in RPY coordinates;

$\mathbf{v}_{\mathrm{NON}}$ is the vector $\mathbf{v}$ expressed in ECI coordinates;

$\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}$ is the coordinate transformation matrix from nonrotating coordinates to rotating coordinates;

and

$$\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}} = \left[\vec{\mathbf{1}}_{x\mathrm{NON}} \quad \vec{\mathbf{1}}_{y\mathrm{NON}} \quad \vec{\mathbf{1}}_{z\mathrm{NON}}\right]. \tag{B.207}$$

The time derivative of $\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}$, as viewed from the nonrotating coordinate frame, can be derived in terms of the dynamics of the unit vectors $\vec{\mathbf{1}}_{x\mathrm{NON}}$, $\vec{\mathbf{1}}_{y\mathrm{NON}}$, and $\vec{\mathbf{1}}_{z\mathrm{NON}}$ in rotating coordinates.

As seen by an observer fixed with respect to the nonrotating coordinates, the nonrotating coordinate directions will appear to remain fixed, but the external inertial reference directions will appear to be changing, as illustrated in Fig. B.14. Gyroscopes fixed in the rotating coordinates would measure three components of the inertial rotation rate vector

$$\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} = \begin{bmatrix} \omega_{x\mathrm{ROT}} \\ \omega_{y\mathrm{ROT}} \\ \omega_{z\mathrm{ROT}} \end{bmatrix} \tag{B.208}$$

in rotating coordinates, but the nonrotating unit vectors, as viewed in rotating coordinates, appear to be changing in the opposite sense, as

$$\frac{d}{dt}\vec{\mathbf{1}}_{x\mathrm{NON}} = -\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes \vec{\mathbf{1}}_{x\mathrm{NON}} \tag{B.209}$$

$$\frac{d}{dt}\vec{\mathbf{1}}_{y\mathrm{NON}} = -\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes \vec{\mathbf{1}}_{y\mathrm{NON}} \tag{B.210}$$

$$\frac{d}{dt}\vec{\mathbf{1}}_{z\mathrm{NON}} = -\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes \vec{\mathbf{1}}_{z\mathrm{NON}}, \tag{B.211}$$

as illustrated in Fig. B.14. The time derivative of the coordinate transformation represented in Eq. B.207 will then be

$$\begin{aligned} \frac{d}{dt}\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}} &= \left[\frac{d}{dt}\vec{\mathbf{1}}_{x\mathrm{NON}} \quad \frac{d}{dt}\vec{\mathbf{1}}_{y\mathrm{NON}} \quad \frac{d}{dt}\vec{\mathbf{1}}_{z\mathrm{NON}}\right] \\ &= \left[-\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes \vec{\mathbf{1}}_{x\mathrm{NON}} \quad -\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes \vec{\mathbf{1}}_{y\mathrm{NON}} \quad -\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes \vec{\mathbf{1}}_{z\mathrm{NON}}\right] \\ &= -\left[\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes\right]\left[\vec{\mathbf{1}}_{x\mathrm{NON}} \quad \vec{\mathbf{1}}_{y\mathrm{NON}} \quad \vec{\mathbf{1}}_{z\mathrm{NON}}\right] \end{aligned} \tag{B.212}$$

$$= -\left[\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes\right]\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}} \tag{B.213}$$

$$\left[\vec{\boldsymbol{\omega}}_{\mathrm{ROT}} \otimes\right] \overset{\mathrm{def}}{=} \begin{bmatrix} 0 & -\omega_{z\mathrm{ROT}} & \omega_{y\mathrm{ROT}} \\ \omega_{z\mathrm{ROT}} & 0 & -\omega_{x\mathrm{ROT}} \\ -\omega_{y\mathrm{ROT}} & \omega_{x\mathrm{ROT}} & 0 \end{bmatrix}. \tag{B.214}$$

The inverse coordinate transformation

$$\mathbf{C}_{\text{NON}}^{\text{ROT}} = \left(\mathbf{C}_{\text{ROT}}^{\text{NON}}\right)^{-1} \tag{B.215}$$

$$= \left(\mathbf{C}_{\text{ROT}}^{\text{NON}}\right)^{T}, \tag{B.216}$$

the transpose of $\mathbf{C}_{\text{ROT}}^{\text{NON}}$, and its derivative

$$\frac{d}{dt}\mathbf{C}_{\text{NON}}^{\text{ROT}} = \frac{d}{dt}\left(\mathbf{C}_{\text{ROT}}^{\text{NON}}\right)^{\text{T}} \tag{B.217}$$

$$= \left(\frac{d}{dt}\mathbf{C}_{\text{ROT}}^{\text{NON}}\right)^{\text{T}} \tag{B.218}$$

$$= \left(-[\vec{\boldsymbol{\omega}}_{\text{ROT}}\otimes]\mathbf{C}_{\text{ROT}}^{\text{NON}}\right)^{\text{T}} \tag{B.219}$$

$$= -\left(\mathbf{C}_{\text{ROT}}^{\text{NON}}\right)^{T}[\vec{\boldsymbol{\omega}}_{\text{ROT}}\otimes]^{\text{T}} \tag{B.220}$$

$$= \mathbf{C}_{\text{NON}}^{\text{ROT}}[\vec{\boldsymbol{\omega}}_{\text{ROT}}\otimes]. \tag{B.221}$$

In the case that ROT is RPY (roll–pitch–yaw coordinates) and NON is ECI (Earth-centered-inertial coordinates), Eq. B.221 becomes

$$\frac{d}{dt}\mathbf{C}_{\text{ECI}}^{\text{RPY}} = \mathbf{C}_{\text{ECI}}^{\text{RPY}}[\vec{\boldsymbol{\omega}}_{\text{RPY}}\otimes], \tag{B.222}$$

and in the case that ROT is ENU (east-north-up coordinates) and NON is ECI (Earth-centered inertial coordinates), Eq. B.213 becomes

$$\frac{d}{dt}\mathbf{C}_{\text{ENU}}^{\text{ECI}} = -[\vec{\boldsymbol{\omega}}_{\text{ENU}}\otimes]\mathbf{C}_{\text{ENU}}^{\text{ECI}}, \tag{B.223}$$

and the derivative of their product

$$\mathbf{C}_{\text{ENU}}^{\text{RPY}} = \mathbf{C}_{\text{ENU}}^{\text{ECI}}\mathbf{C}_{\text{ECI}}^{\text{RPY}} \tag{B.224}$$

$$\begin{aligned}\frac{d}{dt}\mathbf{C}_{\text{ENU}}^{\text{RPY}} &= \left[\frac{d}{dt}\mathbf{C}_{\text{ENU}}^{\text{ECI}}\right]\mathbf{C}_{\text{ECI}}^{\text{RPY}} + \mathbf{C}_{\text{ENU}}^{\text{ECI}}\left[\frac{d}{dt}\mathbf{C}_{\text{ECI}}^{\text{RPY}}\right] \\ &= \left[[-\vec{\boldsymbol{\omega}}_{\text{ENU}}\otimes]\mathbf{C}_{\text{ENU}}^{\text{ECI}}\right]\mathbf{C}_{\text{ECI}}^{\text{RPY}} + \mathbf{C}_{\text{ENU}}^{\text{ECI}}\left[\mathbf{C}_{\text{ECI}}^{\text{RPY}}[\vec{\boldsymbol{\omega}}_{\text{RPY}}\otimes]\right] \\ &= [-\vec{\boldsymbol{\omega}}_{\text{ENU}}\otimes]\underbrace{\mathbf{C}_{\text{ENU}}^{\text{ECI}}\mathbf{C}_{\text{ECI}}^{\text{RPY}}}_{\mathbf{C}_{\text{ENU}}^{\text{RPY}}} + \underbrace{\mathbf{C}_{\text{ENU}}^{\text{ECI}}\mathbf{C}_{\text{ECI}}^{\text{RPY}}}_{\mathbf{C}_{\text{ENU}}^{\text{RPY}}}[\vec{\boldsymbol{\omega}}_{\text{RPY}}\otimes]\end{aligned} \tag{B.225}$$

$$\frac{d}{dt}\mathbf{C}_{\text{ENU}}^{\text{RPY}} = -[\vec{\boldsymbol{\omega}}_{\text{ENU}}\otimes]\mathbf{C}_{\text{ENU}}^{\text{RPY}} + \mathbf{C}_{\text{ENU}}^{\text{RPY}}[\vec{\boldsymbol{\omega}}_{\text{RPY}}\otimes]. \tag{B.226}$$

Equation B.226 was originally used for maintaining vehicle attitude information in strapdown INS implementations, where the variables

$\vec{\omega}_{\text{RPY}}$ = vector of inertial rates measured by the gyroscopes

$$\vec{\omega}_{\text{ENU}} = \vec{\omega}_{\text{earthrate}} + \vec{\omega}_{v_{\text{E}}} + \vec{\omega}_{v_{\text{N}}} \tag{B.227}$$

$$\vec{\omega}_{\oplus} = \omega_{\oplus} \begin{bmatrix} 0 \\ \cos(\phi_{\text{geodetic}}) \\ \sin(\phi_{\text{geodetic}}) \end{bmatrix} \tag{B.228}$$

$$\vec{\omega}_{v_{\text{E}}} = \frac{v_{\text{E}}}{r_{\text{T}} + h} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{B.229}$$

$$\vec{\omega}_{v_{\text{N}}} = \frac{v_{\text{N}}}{r_{\text{M}} + h} \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} \tag{B.230}$$

and

$\omega_{\oplus}$ is earth rotation rate,
ϕ_{geodetic} is geodetic latitude,
v_{E} is the east component of velocity with respect to the surface of the earth,
r_{T} is the transverse radius of curvature of the ellipsoid,
v_{N} is the north component of velocity with respect to the surface of the earth,
r_{M} is the meridional radius of curvature of the ellipsoid (i.e., radius of the osculating circle tangent to the meridian in the meridional plane), and
h is altitude above (+) or below (–) the reference ellipsoid surface (≈mean sea level).

Unfortunately, Eq. B.226 was found to be not particularly well suited for accurate integration in finite-precision arithmetic. This integration problem was eventually solved using quaternions.

B.4.4 Quaternions

The term *quaternions* is used in several contexts to refer to sets of four. In mathematics, it refers to an algebra in four dimensions discovered by the Irish physicist and mathematician Sir William Rowan Hamilton (1805–1865). The utility of quaternions for representing rotations (as points on a sphere in four dimensions) was known before strapdown systems; they soon became the standard representation of coordinate transforms in strapdown systems, and they have since been applied to computer animation.

B.4.4.1 Quaternion Matrices For people already familiar with matrix algebra, the algebra of quaternions can be defined by using an isomorphism between 4×1 **quaternion vectors** q and real 4×4 **quaternion matrices Q**:

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \leftrightarrow \mathbf{Q} = \begin{bmatrix} q_1 & -q_2 & -q_3 & -q_4 \\ q_2 & q_1 & -q_4 & q_3 \\ q_3 & q_4 & q_1 & -q_2 \\ q_4 & -q_3 & q_2 & q_1 \end{bmatrix} \tag{B.231}$$

$$= q_1 Q_1 + q_2 Q_2 + q_3 Q_3 + q_4 Q_4 \tag{B.232}$$

$$Q_1 \overset{\text{def}}{=} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{B.233}$$

$$Q_2 \overset{\text{def}}{=} \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{B.234}$$

$$Q_3 \overset{\text{def}}{=} \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \tag{B.235}$$

$$Q_4 \overset{\text{def}}{=} \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \tag{B.236}$$

in terms of four 4×4 **quaternion basis matrices**, Q_1, Q_2, Q_3, Q_4, the first of which is an identity matrix and the rest of which are antisymmetric.

B.4.4.2 Addition and Multiplication Addition of quaternion vectors is the same as that for ordinary vectors. Multiplication is defined by the usual rules for matrix multiplication applied to the four quaternion basis matrices, the multiplication table for which is given in Table B.1. Note that, like matrix

TABLE B.1. Multiplication of Quaternion Basis Matrices

First Factor	Second Factor			
	Q_1	Q_2	Q_3	Q_4
Q_1	Q_1	Q_2	Q_3	Q_4
Q_2	Q_2	$-Q_1$	Q_4	$-Q_3$
Q_3	Q_3	$-Q_4$	$-Q_1$	Q_2
Q_4	Q_4	Q_3	$-Q_2$	$-Q_1$

multiplication, **quaternion multiplication is noncommutative**; that is, the result depends on the order of multiplication.

Using the quaternion basis matrix multiplication Table B.1, the ordered product **AB** of two quaternion matrices

$$\mathbf{A} = a_1 Q_1 + a_2 Q_2 + a_3 Q_3 + a_4 Q_4 \tag{B.237}$$

$$\mathbf{B} = b_1 Q_1 + b_2 Q_2 + b_3 Q_3 + b_4 Q_4 \tag{B.238}$$

can be shown to be

$$\begin{aligned}\mathbf{AB} = {} & (a_1 b_1 - a_2 b_2 - a_3 b_3 - a_4 b_4) Q_1 + (a_2 b_1 + a_1 b_2 - a_4 b_3 + a_3 b_4) Q_2 \\ & + (a_3 b_1 + a_4 b_2 + a_1 b_3 - a_2 b_4) Q_3 + (a_4 b_1 - a_3 b_2 + a_2 b_3 + a_1 b_4) Q_4\end{aligned} \tag{B.239}$$

in terms of the coefficients a_k, b_k and the quaternion basis matrices.

B.4.4.3 Conjugation Conjugation of quaternions is a unary operation analogous to conjugation of complex numbers, in that the real part (the first component of a quaternion) is unchanged and the other parts change sign. For quaternions, this is equivalent to transposition of the associated quaternion matrix

$$\mathbf{Q} = q_1 Q_1 + q_2 Q_2 + q_3 Q_3 + q_4 Q_4, \tag{B.240}$$

so that

$$\mathbf{Q}^{\mathrm{T}} = q_1 Q_1 - q_2 Q_2 - q_3 Q_3 - q_4 Q_4 \tag{B.241}$$

$$\leftrightarrow \mathbf{q}^* \tag{B.242}$$

$$\mathbf{Q}^{\mathrm{T}} \mathbf{Q} = \left(q_1^2 + q_2^2 + q_3^2 + q_4^2\right) Q_1 \tag{B.243}$$

$$\leftrightarrow \mathbf{q}^* \mathbf{q} = |\mathbf{q}|^2. \tag{B.244}$$

B.4.4.4 Representing Rotations The problem with rotation vectors as representations for rotations is that the rotation vector representing successive rotations $\vec{\boldsymbol{\rho}}_1, \vec{\boldsymbol{\rho}}_2, \vec{\boldsymbol{\rho}}_3, \ldots, \vec{\boldsymbol{\rho}}_n$ is not a simple function of the respective rotation vectors.

This representation problem is solved rather elegantly using quaternions, such that the quaternion representation of the successive rotations is represented by the quaternion product $\mathbf{q}_n \times \mathbf{q}_{n-1} \times \mathbf{q}_3 \times \mathbf{q}_2 \times \mathbf{q}_1$; that is, each successive rotation can be implemented by a single quaternion product.

The quaternion equivalent of the rotation vector $\vec{\boldsymbol{\rho}}$ with $|\vec{\boldsymbol{\rho}}| = \theta$,

$$\vec{\boldsymbol{\rho}} \overset{\text{def}}{=} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \end{bmatrix} \overset{\text{def}}{=} \theta \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}, \tag{B.245}$$

(i.e., where **u** is a unit vector) is

$$\mathbf{q}(\vec{\rho}) \stackrel{\text{def}}{=} \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ \rho_1 \sin\left(\frac{\theta}{2}\right)\Big/\theta \\ \rho_2 \sin\left(\frac{\theta}{2}\right)\Big/\theta \\ \rho_3 \sin\left(\frac{\theta}{2}\right)\Big/\theta \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ u_1 \sin\left(\frac{\theta}{2}\right) \\ u_2 \sin\left(\frac{\theta}{2}\right) \\ u_3 \sin\left(\frac{\theta}{2}\right) \end{bmatrix}, \tag{B.246}$$

and the vector **w** resulting from the rotation of any three-dimensional vector

$$\mathbf{v} \stackrel{\text{def}}{=} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

through the angle θ about the unit vector **u** is implemented by the quaternion product

$$\mathbf{q}(w) \stackrel{\text{def}}{=} \mathbf{q}(\vec{\rho})\mathbf{q}(\mathbf{v})\mathbf{q}^*(\vec{\rho}) \tag{B.247}$$

$$\stackrel{\text{def}}{=} \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ u_1 \sin\left(\frac{\theta}{2}\right) \\ u_2 \sin\left(\frac{\theta}{2}\right) \\ u_3 \sin\left(\frac{\theta}{2}\right) \end{bmatrix} \times \begin{bmatrix} 0 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix} \times \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ -u_1 \sin\left(\frac{\theta}{2}\right) \\ -u_2 \sin\left(\frac{\theta}{2}\right) \\ -u_3 \sin\left(\frac{\theta}{2}\right) \end{bmatrix} \tag{B.248}$$

$$= \begin{bmatrix} 0 \\ w_1 \\ w_2 \\ w_3 \end{bmatrix} \tag{B.249}$$

$$w_1 = \cos(\theta)v_1 + [1-\cos(\theta)][u_1(u_1v_1 + u_2v_2 + u_3uv_3)] + \sin(\theta)[u_2v_3 - u_3v_2] \tag{B.250}$$

$$w_2 = \cos(\theta)v_2 + [1-\cos(\theta)][u_2(u_1v_1 + u_2v_2 + u_3uv_3)] + \sin(\theta)[u_3v_1 - u_1v_3] \tag{B.251}$$

$$w_3 = \cos(\theta)v_3 + [1-\cos(\theta)][u_3(u_1v_1 + u_2v_2 + u_3uv_3)] + \sin(\theta)[u_1v_2 - u_2v_1], \tag{B.252}$$

or

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \mathbf{C}(\vec{\boldsymbol{\rho}}) \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}, \tag{B.253}$$

where the rotation matrix $\mathbf{C}(\vec{\boldsymbol{\rho}})$ is defined in Eq. B.118 and Eq. B.247 implements the same rotation of $\mathbf{v}$ as the matrix product $\mathbf{C}(\vec{\boldsymbol{\rho}})\mathbf{v}$. Moreover, if

$$\mathbf{q}(\mathbf{w}_0) \stackrel{\text{def}}{=} \mathbf{v} \tag{B.254}$$

and

$$\mathbf{q}(\mathbf{w}_k) \stackrel{\text{def}}{=} \mathbf{q}(\vec{\boldsymbol{\rho}}_k)\mathbf{q}(\mathbf{w}_{k-1})\mathbf{q}^*(\vec{\boldsymbol{\rho}}_k) \tag{B.255}$$

for $k = 1, 2, 3, \ldots, n$, then the nested quaternion product

$$\mathbf{q}(\mathbf{w}_n) = \mathbf{q}(\vec{\boldsymbol{\rho}}_n)\cdots\mathbf{q}(\vec{\boldsymbol{\rho}}_2)\mathbf{q}(\vec{\boldsymbol{\rho}}_1)\mathbf{q}(\mathbf{v})\mathbf{q}^*(\vec{\boldsymbol{\rho}}_1)\mathbf{q}^*(\vec{\boldsymbol{\rho}}_2)\cdots\mathbf{q}^*(\vec{\boldsymbol{\rho}}_n) \tag{B.256}$$

implements the succession of rotations represented by the rotation vectors $\vec{\boldsymbol{\rho}}_1$, $\vec{\boldsymbol{\rho}}_2$, $\vec{\boldsymbol{\rho}}_3$, . . . , $\vec{\boldsymbol{\rho}}_n$, and the single quaternion

$$\mathbf{q}_{[n]} \stackrel{\text{def}}{=} \mathbf{q}(\vec{\boldsymbol{\rho}}_n)\mathbf{q}(\vec{\boldsymbol{\rho}}_{n-1})\cdots\mathbf{q}(\vec{\boldsymbol{\rho}}_3)\mathbf{q}(\vec{\boldsymbol{\rho}}_2)\mathbf{q}(\vec{\boldsymbol{\rho}}_1) \tag{B.257}$$

$$= \mathbf{q}(\vec{\boldsymbol{\rho}}_n)\mathbf{q}_{[n-1]} \tag{B.258}$$

then represents the net effect of the successive rotations as

$$\mathbf{q}(\mathbf{w}_n) = \mathbf{q}_{[n]}\mathbf{q}(\mathbf{w}_0)\mathbf{q}_{[n]}^*. \tag{B.259}$$

The initial value $\mathbf{q}_{[0]}$ for the rotation quaternion will depend upon the initial orientation of the two coordinate systems. The initial value

$$\mathbf{q}_{[0]} \stackrel{\text{def}}{=} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{B.260}$$

applies to the case that the two coordinate systems are aligned. In strapdown system applications, the initial value $\mathbf{q}_{[0]}$ is determined during the INS alignment procedure.

Equation B.257 is the much-used quaternion representation for successive rotations, and Eq. B.259 is how it is used to perform coordinate transformations of any vector $\mathbf{w}_0$.

This representation uses the four components of a unit quaternion to maintain the transformation from one coordinate frame to another through a succession of rotations. In practice, computer roundoff may tend to alter the magnitude of the allegedly unit quaternion, but it can easily be rescaled to a unit quaternion by dividing by its magnitude.

B.5 NEWTONIAN MECHANICS IN ROTATING COORDINATES

Using Eqs. B.117 and B.118 derived above, one can easily derive formulas for the corrections to Newtonian mechanics in rotating coordinate systems. For that purpose, we substitute the rotation vector

$$\boldsymbol{\rho} = -\boldsymbol{\omega} t, \tag{B.261}$$

where t is time and $\boldsymbol{\omega}$ is the rotation rate vector of the rotating coordinate system with respect to the nonrotating coordinate system. There is a negative sign on the right-hand side of Eq. B.261 because the rotation matrix of Eqs. B.117 and B.118 represents the physical rotation of a coordinate system, whereas the application here represents the coordinate transformation from the unrotated (i.e., inertial) coordinate system to the rotated coordinate system.

B.5.1 Rotating Coordinates

Let x_{ROT}, $\dot{x}_{\mathrm{ROT}}$, $\ddot{x}_{\mathrm{ROT}}$, . . . represent the position, velocity, acceleration, and so on, of a point mass in the rotating (but nonaccelerating) coordinate system, rotating at angular rate $|\boldsymbol{\omega}|$ about a vector $\boldsymbol{\omega}$; that is, the components of $\boldsymbol{\omega}$ are the components of rotation rate about the coordinate axes, and the axis of rotation passes through the origin of the rotating coordinate system.

In order to relate these dynamic variables to Newtonian mechanics, let x_{NON}, $\dot{x}_{\mathrm{NON}}$, $\ddot{x}_{\mathrm{NON}}$, . . . represent the position, velocity, acceleration, and so on, of the same point mass in an inertial (i.e., nonrotating) coordinate system coincident with the rotating system at some arbitrary reference time t_0.

Then the coordinate transform from inertial to rotating coordinates can be represented in terms of a rotation matrix:

$$x_{\mathrm{ROT}}(t) = C_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)\, x_{\mathrm{NON}}(t) \tag{B.262}$$

$$\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t) = \frac{\boldsymbol{\omega}\boldsymbol{\omega}^T}{|\boldsymbol{\omega}|^2} - \frac{\cos(|\boldsymbol{\omega}|(t-t_0))}{|\boldsymbol{\omega}|^2}[\boldsymbol{\omega}\otimes][\boldsymbol{\omega}\otimes] - \frac{\sin(|\boldsymbol{\omega}|(t-t_0))}{|\boldsymbol{\omega}|}[\boldsymbol{\omega}\otimes] \tag{B.263}$$

$$= \cos(|\boldsymbol{\omega}|(t-t_0))\mathbf{I}_3 + \frac{[1-\cos(|\boldsymbol{\omega}|(t-t_0))]}{|\boldsymbol{\omega}|^2}\boldsymbol{\omega}\boldsymbol{\omega}^T - \frac{\sin(|\boldsymbol{\omega}|(t-t_0))}{|\boldsymbol{\omega}|}[\boldsymbol{\omega}\otimes]. \tag{B.264}$$

B.5.2 Time Derivatives of Matrix Products

Leibniz rule for derivatives of products applies to vector-matrix products, as well. Using this rule, the time derivatives of $x_{\mathrm{ROT}}(t)$ can be expressed in terms of the time derivatives of $\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)$ and $x_{\mathrm{ROT}}(t)$ as

$$\dot{x}_{\mathrm{ROT}}(t) = \dot{\mathbf{C}}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)\, x_{\mathrm{NON}}(t) + \mathbf{C}_{\mathrm{NON}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)\, \dot{x}_{\mathrm{ROT}}(t) \tag{B.265}$$

$$\ddot{x}_{\mathrm{ROT}}(t) = \ddot{\mathbf{C}}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)\, x_{\mathrm{NON}}(t) + 2\dot{\mathbf{C}}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)\, \dot{x}_{\mathrm{NON}}(t) + \mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)\, \ddot{x}_{\mathrm{NON}}(t). \tag{B.266}$$

The first and second time derivatives of $\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t)$ can be derived by straightforward differentiation:

$$\dot{\mathbf{C}}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t) = \frac{\sin(|\boldsymbol{\omega}|(t-t_0))}{|\boldsymbol{\omega}|}[\boldsymbol{\omega}\otimes][\boldsymbol{\omega}\otimes] - \cos(|\boldsymbol{\omega}|(t-t_0))[\boldsymbol{\omega}\otimes] \tag{B.267}$$

$$\ddot{\mathbf{C}}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t) = \cos(|\boldsymbol{\omega}|(t-t_0))[\boldsymbol{\omega}\otimes][\boldsymbol{\omega}\otimes] + |\boldsymbol{\omega}|\sin(|\boldsymbol{\omega}|(t-t_0))[\boldsymbol{\omega}\otimes]. \tag{B.268}$$

B.5.3 Solving for Centrifugal and Coriolis Accelerations

Evaluated at $t = t_0$, these become

$$\mathbf{C}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t_0) = I_3 \tag{B.269}$$

$$\dot{\mathbf{C}}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t_0) = -[\boldsymbol{\omega}\otimes] \tag{B.270}$$

$$\ddot{\mathbf{C}}_{\mathrm{ROT}}^{\mathrm{NON}}(\boldsymbol{\omega}, t_0) = [\boldsymbol{\omega}\otimes][\boldsymbol{\omega}\otimes], \tag{B.271}$$

and the corresponding time derivatives at $t = t_0$ are

$$x_{\mathrm{ROT}}(t_0) = x_{\mathrm{NON}}(t_0) \tag{B.272}$$

$$\dot{x}_{\mathrm{ROT}}(t_0) = -[\boldsymbol{\omega}\otimes]\, x_{\mathrm{NON}}(t_0) + \dot{x}_{\mathrm{NON}}(t_0) \tag{B.273}$$

$$= -\underbrace{[\boldsymbol{\omega}\otimes]\, x_{\mathrm{ROT}}(t_0)}_{\text{TANGENTIAL}} + \underbrace{\dot{x}_{\mathrm{NON}}(t_0)}_{\text{INERTIAL}} \tag{B.274}$$

$$\dot{x}_{\mathrm{NON}}(t_0) = \dot{x}_{\mathrm{ROT}}(t_0) + [\boldsymbol{\omega}\otimes]\, x_{\mathrm{ROT}}(t_0) \tag{B.275}$$

$$\ddot{x}_{\mathrm{ROT}}(t_0) = [\boldsymbol{\omega}\otimes][\boldsymbol{\omega}\otimes]\, x_{\mathrm{ROT}}(t_0) - 2[\boldsymbol{\omega}\otimes]\{\dot{x}_{\mathrm{ROT}}(t_0) + [\boldsymbol{\omega}\otimes]\, x_{\mathrm{ROT}}(t_0)\} + \ddot{x}_{\mathrm{NON}}(t_0) \tag{B.276}$$

$$= -\underbrace{[\boldsymbol{\omega}\otimes][\boldsymbol{\omega}\otimes]\, x_{\mathrm{ROT}}(t_0)}_{\text{CENTRIFUGAL}} - \underbrace{2[\boldsymbol{\omega}\otimes]\, \dot{x}_{\mathrm{ROT}}(t_0)}_{\text{CORIOLIS}} + \underbrace{\ddot{x}_{\mathrm{NON}}(t_0)}_{\text{SPECIFIC FORCE}}, \tag{B.277}$$

where the labels under the various terms in the formula for acceleration observed in rotating coordinates identify the *centrifugal*, *Coriolis*, and *specific-*

force accelerations in rotating coordinates. The *specific-force* acceleration is that due to inertial forces applied to the point mass. The others are artifacts due to rotation.

In similar fashion, the velocity term in rotating coordinates labeled "INERTIAL" is the point mass velocity in inertial coordinates, and that labeled "TANGENTIAL" is an artifact due to rotation.

Because the reference time t_0 was chosen arbitrarily, these formulas apply for all times.

INDEX

621B Program, 474

Abbott, Anthony, 475
Acceleration, 55
Accelerometer, 55
  calibration, 63, 87
  electromagnetic, 27
  gyroscopic, 16, 21, 27
  integrating, 15–16
  mass-spring, 28
  MEMS, 28
  pendulous, 66
  proof mass, 27
Adaptive Kalman filter, 384, 393
Aerospace Corp., 474–475
A-GPS, 199, 235–240
  3GPP, 236
AIRS, 80
Alignment (INS), 58, 86
  GNSS-aided, 85
  gyrocompass, 58, 85–87, 99
  optical, 67, 84
  transfer, 85
Almanac
  GLONASS, 143
  GPS, 113
Alpha wander (INS), 81, 516
Altimeter, 68, 90, 101
  barometric, 454
    model, 455–456
Altitude, 58
  orthometric, 511
  stabilization, 90
Ambiguity resolution, 227
Angus, John, xxxi
Anomaly
  eccentric, 119
  mean, 120
  true, 119
Antenna,
  adaptive
    space-time (STAP), 185
    steerable, 180
  anti-jamming, 181
  axial ratio, 156–157, 173–174
  bandwidth, 153, 161, 173
  beamforming, 182
  calibration, 187–188

Global Navigation Satellite Systems, Inertial Navigation, and Integration, Third Edition.
Mohinder S. Grewal, Angus P. Andrews, and Chris G. Bartone.

Antenna (*cont'd*)
choke-ring, 176
multipath rejection, 177
computational model, 164
degrees of freedom, 180–185
design, 152
analysis, 152
CEMs, 164
dipole, 166
directivity, 159
efficiency, 159
electronics, 181, 186
gain, 159
impedance, 160, 169–170
model
computational, 164
noise figure, 163–164
noise temperature, 163–164
patch, 166–176
multifrequency, 175–176
probe-fed, 170–176
single-frequency, 168–174
phased-array, 180–187
adaptable, 185–187
CRPA, 180–186
pinwheel, 179
planar based
advanced, 177
polarization, 156–159, 173
Q, 173–174
radiation pattern, 155
reference point, 187
return loss, 160–161
SFAP, 185
space-adaptive, 180–185
spiral, 178
STAP, 185
steerable, 180
survey grade, 176–179
SWR, 160–164, 169
Anti-jamming, 124, 130, 134
Anti-spoofing, 130
A posteriori, 353
APL, 474
A priori, 353
Argmax, 355
Argument of latitude, 118, 507
Argument of perigee, 117
ARNS, 134
Ascending node, 507
Ascent plane, 518–519
Ash, Michael, xxxi
Attitude, 57, 79, 84, 90
integration, 55, 58
quaternion representation, 95
rates, 58
sensor, 68
strapdown, 90
Autocorrelation, 125
Autonetics, 22, 475

Black box model, 59
BD (BeiDou), 146
BeiDou, 10, 146
Bierman, Gerald J., 497
Birnbaum, Mel, 475
Blaser, Herbert, xxxi
Bohnenberger, Johann, 14, 17
Boltinghouse, Helen, xxxi
Boltinghouse, Joseph C., 21–22
Bortz, John E., 93
Bortz rate vector, 93
Boykow, Johann M., 16
BPF, 200
BPSK, 109, 111–112, 143, 147
Brasch, Michael, xxxi
Brooks, David, xxxi
Brooks, Robert, 366
Brownian motion, 366
Bryan effect, 26
Bucy, Richard S., 395

Calibration, 56, 63–66
accelerometer, 87
antenna, 187
gyroscope, 80
instability, 66
parameters, 65
Carlson, Neal A., 497
Carouseling (INS), 81
strapdown, 82
Carrier tracking, 213
Celestial sphere, 505
CEM, 164–166
Centrifugal acceleration, 66, 440, 446, 549

CEP, xxxiv, 29, 101
  rate, 101
    versus sensor noise, 461
Cheung, Laura A., xxxi
CIGTIF, 103
Circular error probable, 29, 101
Cockpit displays, 90
Code tracking, 213
Compass/BeiDou, 10, 146
  datum, 146
  orbits, 146
  signal structure, 146
  time, 146
Computational electronic model, 164
Computer, 99
  navigation, 57, 99
    operating system, 100
    requirements, 100
Coning motion, 90–93
  correction, 93–98
Consolidated Vultee Aircraft, 20
Control, 2
Coordinates, 432, 500
  alpha wander, 516
  Cartesian, 504
  celestial, 505
  ECEF, 41, 69, 508, 516
  ECI, 69, 504
  ENU, 41, 69, 515
  Euclidean, 504
  GNSS, 35, 521–522
  inertial, 55, 69, 503
  INS analysis, 432
  Keplerian, 505
  locally level, 55, 69, 515
  LTP, 69, 515
  navigation, 55, 69
  NED, 69, 516
  polar, 504
  rotating, 540
  RPY, 69, 518
  SAE, 518
  terrestrial, 69
  transformations, 500
Coordinate transformation, 500
  composition rule, 503
  ENU/ECEF, 41, 516
  ENU/NED, 516
  matrices, 500
  NED/ECEF, 516
  notation, 500
  RPY/ENU, 519
  RPY/NED, 519, 523
Core variables, 432, 434
  error model, 481
  history, 10–30
  horizontal, 459
  initialization, 84
  instability, 452
  location errors, 434, 437
  misalignments, 433–436
  orientation errors, 433
  parameters, 433
  performance, 101–102
  Schuler oscillation, 457
  sensor compensation, 481
  sensor noise, 459
  signal processing, 89, 97–99, 439
    hardware, 28
  solution, 432
  strapdown, 28
  tilt error, 434
  variables, 433
  vertical channel, 452
Corey, Randall, xxxi
Coriolis
  acceleration, 440, 445, 549
  effect, 90, 103
  gyroscope, 24–27
Correlate, 206
Correlation number, 379
Cosine rule, 502
Covariance matrix, 352, 356
  propagation, 353
  update, 363, 365
Cross-correlation, 129
CRPA, 180–186
Cruise applications (INS), 80
CSDL, 20, 26
Curtiss-Wright Corp., 20
Curvature
  east-west, 513
  ellipsoidal, 515
  meridional, 515
  north-south, 515
  transverse, 513

Darwin, George, 70
Data stripping, 240
Dead reckoning, 3
Declination, 505
Delco Electronics, 81
Descartes, René, 504
Descending node, 507
Despreading, 127, 206, 214
Detection confirmation, 211
Detection threshold, 207–208
DeVries, Thomas W., 475
Direction cosines, 96, 502, 549
Distribution matrix
  dynamic noise, 366
  sensor noise, 361
Doppler shift, 206–207, 209, 213, 228–229
Dot product, 502
Draper, C. Stark, 10, 19, 54
Dual-frequency, 131, 133–134, 136
Dynamic coupling, 439
  acceleration, 449
  centrifugal acceleration, 440, 446
  compensation errors, 439
  Coriolis acceleration, 440, 445
  earthrate leveling, 440, 447
  gravity calculation, 440, 443–445
  matrix, 441
    navigation errors, 439
    partitioning, 441, 443
    velocity integration, 440, 442
    velocity leveling, 440, 448
Dynamic disturbance, 366
  distribution matrix, 366

E911, 235
Earth
  flattening, 433
  geoid, 513
  gravity constant, 433
  models, 68
  radius
    geocentric, 514
    mean, 433
  symbol, 433
Eccentric anomaly, 119–122
Eccentricity, 117–118
Ephemeris
  GLONASS, 142
  GPS, 113, 36
Error budgets, 288
ESA, 144

False alarm probability, 208
FDMA, 142
Filter
  BPF, 164, 167
  FIR, 180
  Kalman, 30, 87, 350, 477
Forward model, 351
Forward solution, 371
Foucault, J. B. Léon, 14
Foucault pendulum, 458
Free inertial navigation, 101
Frequency bin, 207
FSLF, 132
Fully coherent, 239

Galileo, 7–10, 144–146
  commercial service, 8
  data, 145
  GIOVE, 144
  navigation data formats, , 145–146
  open service, 7
  orbits, 36–37, 145
  public regulated service, 8
  safety of life, 7
  search and rescue, 8
  services, 145
  signal structure, 145–146
Gamow, George, 19
Gaussian
  distribution , 355
  likelihood, 357
GDOP, 41, 44–45, 497
Geocentric radius, 514
Geodesy, 509
Geoid, 72–73
Gimbal, 57, 77–80
  advantages, 79
  disadvantages, 80
  Euler angles, 57
  flip, 81
  lock, 79, 524–525
Glen L. Martin Co., 20
Global search, 212
GLONASS, xxxv, 6–7, 142–144
  altitude, 142
  blocks, 143

carrier structure, 143
CDMA, 144
data format, 143
ephemeris, 142
FDMA, 142
HPNS, 143
K, 7, 143
M, 7, 143
modernization, 144
CDMA, 144
orbits, 6, 142
signal structure, 6, 142–144
SPNS, 143
spreading codes, 143
GNSS, 4, 35, 250, 472
carrier phase
windup, 188
error model, 479
navigation, 36, 472
holy point, 490
performance, 485
overview, 4
receiver, 351
acquisition and tracking, 204–223
ADC, 203
AGC, 203
aiding inputs, 198–199
applications, 193–199
architecture, 199–204
baseband processing, 204
carrier Doppler measurement, 228
carrier tracking loop, 218–222
clock bias, 226
clock model, 479
code tracking, 213–217
code tracking loop, 213–218
coherent tracking, 217
Costas PLL, 218–219
data bit demodulation, 223
data bit synchronization, 222
differential, 196–198
discriminator—carrier, 219, 221–222
discriminator—code, 214–216
DLL, 215
dual-frequency, 194
early, 215
early minus late, 215
FLL, 222
I, 207
IF, 199–202
integrated Doppler, 218–220, 223, 229–231
late, 215
loop bandwidth, 217
loop filter, 217
NCO-carrier, 214, 218–219
NCO-code, 214–217
noncoherent tracking, 217
PDI, 215
PLL, 218–221
pseudorange, 224–226
punctual correlation, 215
Q, 207
RF front-end, 199–201
searching, 206–210
signal detection confirmation, 210–212
SNR, 202–203
satellite position, 117–122
signal structure, 108
Compass/BD, 146
Galileo, 145–146
GLONASS, 142–144
GPS, 116–141
QZSS, 147
GNSS data errors, 250–262
ephemeris, 250
ephemeris errors, 285
onboard clock, 285
ionosphere, 250
GNSS SBAS, 254
Klobuchar, 252
propagation, 251
using pseudorange, 262
multipath, 264
causing range error, 264
limits of MT mitigation, 283
mitigation, 266
MMT technology, 272
problem, 264
spatial domain, 267
time domain, 269
receiver error models
continuous, 287
discrete, 287
receiver errors, 250
satellite clock, 250

GNSS data errors (*cont'd*)
troposphere
propagation, 263
GNSS/INS integration, 30, 472
antenna offset, 478, 490
applications, 31
dynamic coupling, 481–483
dynamic simulation, 485
early history, 473–475
implementation, 31
loosely coupled, 475–477
measurement sensitivity, 484
overview, 30, 473
performance, 489
process noise, 480, 484
receiver clock model, 479
state transition matrix, 485
state variables, 482
tightly coupled, 475–477
trajectories, 491
ultra tightly coupled, 477
unified model, 477–484
GNSS receiver
antenna
bandwidth, 153
design, 152
patch, 166–176
performance, 152
polarization, 156
radiation pattern, 155
survey grade, 176
Goddard, Robert H., 15
Gold codes, 125, 146, 150
Gow flip, 81
GPS, 2, 4–6
II, 6
III, 6, 141–142
acquisition, 124
almanac, 113
anti-jamming, 124, 130
assisted, 199, 235–240
high sensitivity, 235–238
blocks, 141
C/A code, 109, 123
autocorrelation, 125
multipath, 123
civic code, 133, 136
civil signal, 136, 140
clock error, 122–123
code structure, 109–112
subframes, 116–141
ephemeris, 113, 116–119
history, 4, 473–475
L1, 109
L1C, 140–141
L2, 109
L2C, 135–136
L3, 144
L5, 135, 137–139
M-code, 135, 139
modernization, 133–142
spectrum, 135
orbits, 4, 36–37
position calculation, 48, 117
algorithm, 118
power, 132
propagation delays, 5
P(Y) code, 115, 129–130
receiver
channels, 195
code selection, 195
satellite position, 48, 118–122
selective availability, 5, 379
signal, 4–5
carriers, 131
despreading, 127–129
signal power
received, 133
transmitted, 132
signal spreading, 126
PSD, 126
subfames, 116–141
temporal, 124
transmitted power levels, 132
GPS/INS integration, 475
GPSoft, 458
toolboxes, 495
Gravity, 55, 57, 68–71
model, 443
potential, 71
Gravity Probe B, 22
Guidance, 2
Guier, William H., 474
Gyro, 55
Gyrocompassing, 58, 99
accuracy, 86

Gyroscope, 14, 55
  calibration, 80
  Coriolis, 21
    vibrating, 13, 24
  displacement, 56
  electrostatic, 22
  fiber optic, 24
  gas bearing, 14, 21
  hemispherical resonator, 26
  MEMS, 26
  momentum wheel, 14, 21
  optical, 21
  rate, 56
  ring laser, 23
  tuning fork, 25
  whole-angle, 56
  wine glass, 25
  Zero Lock, 23

Harmonic resonator, 366
Heading, 519
Holy point, 490
Honeywell Corporation, 22, 27
Horne, James, xxxi
Host vehicle, 58
  dynamic models, 400
HOW, 114
HRG, 26, 61
HVAC, 61

ICBM, 29
IFOG, 61
IMU, 57
  misalignments, 434
  mounting, quasi-rigid, 78
Indexing (INS), 81–82
  gimbaled, 81
  Gow flip, 81
  strapdown, 82
Inertia, 55
Inertial
  coordinates, 55
  measurement unit, 56, 57
  navigation system, 57
  platform, 56, 78
  reference frame, 55
  reference unit, 57
  sensor, 55
    accelerometer, 55
    assembly, 56
    black box model, 59
    calibration, 56, 63–66
    compensation, 63–64
    compensation error, 461
    error model, 59
    gyroscope, 55
    input axis, 56
    misalignments, 63–65
    multi-axis, 56
    noise, 459, 461
Inertial grade (sensor), 13, 15, 16
Inertial navigation, 3, 10, 472, 488
  accuracy, 29
  computer, 99–101
  error analysis, 430
Inertial platform, 15–16
Inertial sensors, 11, 13, 21
  inertial grade, 13, 15–16
  requirements, 12
Information matrix, 357
Initialization (INS), 58
Innovations, 375
Input axis, 56
INS, 57
  accuracy, 102
  alignment, 58
  altitude stabilization, 68, 90, 101
  carouseled, 81
  cruise applications, 80
  error analysis, 430
  field test, 103
  floated, 77
  free inertial, 101
  gimbaled, 56–58
  gyrocompassing, 58
  holy point, 490
  host vehicle, 58
  implementation, 57
    gimbaled, 57, 89
    strapdown, 57, 97–99
  indexing, 81
  initialization, 58
  leveling, 58
  performance, 101
    stand-alone, 102

INS (*cont'd*)
self-alignment, 58
self-calibration, 80
software, 100
strapdown, 97–99
testing, 102
user interface, 57
vertical channel, 68, 90
stabilization, 68, 90, 101
vibration isolators, 58, 78
Instability, 452
INS errors, 452
altimeter aiding, 454
Instrument cluster, 56
Integrated GNSS/INS, 472
Interface specification, 109, 150–151
Interference suppression, 128
Ionospheric delay data (GPS), 113–114
ISA, 56
GNSS antenna offset, 490
misalignment, 434

JAXA, 147

Kailath, Thomas, 422
Kalman filter, 30, 87, 350, 477
adaptive, 384, 393
equations, 376
extended, 392
Kalman-Bucy, 395
linearized, 391
Kalman gain, 354–364
Kalman-Bucy filter, 395
Kepler equation, 121
Kepler, Johannes, 505
Keplerian parameters, 496, 505

L1–2, 109
L3, 144
L5, 137–139
LAMBDA, 227
Langevin equation, 366
Langley, Richard B., xxxi
Laplace distribution, 356
Latitude, 509
argument of, 118, 507
geocentric, 509, 513
geodetic, 72, 510, 514
geometric, 509
orthometric, 72
parametric, 509, 514
LEM, 29
Leveling, 440
Likelihood function, 356
Gaussian, 357
Linearized Kalman filter, 391
LNA, 200–201
Local search, 212
Local tangent plane, 515
Locally level coordinates, 55, 69, 515
Location errors, 434
Longitude, 509
Longitude of ascending node, 117–118
low noise amplifier, 164, 167
LSB, 61
LTP coordinates, 69, 515
Lukesh, John, 475

Magnavox, 474, 475
Martin, Edward H., xxxi, 475
Mask angle, 205
MATLAB®, 97
Matrix
coordinate transformation, 500
derivatives, 534, 549
direction cosines, 539
exponential, 371
notation, 500
pseudoinverse, 363
rotation, 525, 527
Maximum likelihood
gain, 354–364
measurement estimation, 355, 362
carrier phase, 233–234
code phase, 231–233
frequency, 233–234
McClure, Frank, 474
Mean anomaly, 117–120
Mean sea level, 71
Measurement, 360, 373
likelihood, 361
matrix, 360, 375
noise, 354, 360–361
distribution matrix, 361
predicted, 375
update, 353
vector, 360
MEMS, 26, 28

MEO, 207
Meridional curvature, 514
Microstrip antenna, 168
Misalignments (INS), 434
Missed detection, 208, 210
MMIA, 16
Moore-Penrose inverse, 360
Mueller, Fritz, 16–19
Multipath (GNSS), 123, 134, 176
MWG, 14, 21

Nautical mile, 101–102
NAV data, 112–117
Navigation, 1
  celestial, 3
  computer, 57, 99
    operating system, 100
  coordinates, 55, 58, 69, 84
  error (INS)
    centrifugal, 440, 446
    core variables, 432
    Coriolis, 440, 445
    dynamics, 441–459
    earthrate, 440, 447
    gravity, 440, 444
    INS, 432
    location, 434
    variables, 434
    velocity integration, 442
    velocity leveling, 448
  free inertial, 101
    performance, 102
  GNSS, 35, 472
  inertial, 3, 54, 97–99
    error analysis, 430
  model
    horizontal, 459
    nine-state, 432
    seven-state, 459
    unified, 477
  modes, 2
  pure inertial, 101
  radio, 3
  software, 99
  solution, 84, 432, 433
    core variables, 432
  technology, 1
Near-far problem, 243
Nease, Robert F., xxxi
NED coordinates, 516
Newton Isaac, 10, 55, 70, 365
Nine core variables (INS), 432
Noise, 361
  distribution matrix, 366, 374
  dynamic disturbance, 366
  fixed-pattern, 61–62
  measurement, 361
  observation, 377
  plant, 377
  process, 353
  sensor, 60–62, 360
  white, 365
Nordsieck, Arnold T., 22
Normal distribution, 356
North American Aviation, 20
Northing, 516
Northrup Corp., 20, 475
Nuisance variables, 351–352

Observation
  matrix, 375
  noise, 377
  update, 353
Orientation errors, 434
Orthometric height, 511
OS, 7

Partial coherent, 239
Patch antenna, 166–176
P-code, 130
Perigee, 120
  argument of, 117, 507
PIGA, 16
Pilotage, 2
Pitch angle, 519
Pitch plane, 518
Plant noise, 377
Podkorytov, Andrey, xxxi
Polarization
  antenna, 156
  circular
    right/left, 157, 173
  linear, 157
Poon, Samantha, xxxi
Potter, James E., 418
Potter square-root filter, 497
Power spectrum, 126
PPS, 109

PRN, 109
  codes, 4
Probability
  of detection, 208
  of false alarm, 208
Process noise, 353, 366
Proof mass, 27
PRS, 8
Pseudoinverse, 360
Pseudolite, 243–244
Pseudorange, 224–226, 387, 480, 521
  error model, 480–481
  measurement sensitivity, 484
  noise model, 480
P(Y)-code, 115, 129

Quasi-rigid (IMU mounting), 78
Quaternion, 95, 543
  addition, 544
  integration, 96
  multiplication, 544
  rotations, 545
  transformations, 97
QZSS, 147
  L1-LEX, 147
  L1-SAIF, 147
  orbits, 147
  signal structure, 147

RA, 499, 505
RAAN, 507
Radio navigation, 3–4, 15
Random walk, 60, 378
Receiver antenna/ISA offset, 490
Republic Aviation, 20
Riccati, Jacopo F., 395
Riccati equation, 353–354
  steady-state, 401–404
Right ascension, 505
RLG, 23, 61
Roll angle, 519
Roll-pitch-yaw coordinates, 518
Rotation
  derivatives, 529
  matrix, 525, 527
  mechanics, 548
  quaternion, 545
  rate vector, 89, 94
  vector, 524

RTCM, 197–198
RTOS, 100

SA, 5, 288
SAE coordinates, 518
Sagnac effect, 21, 24
SAR, xxxviii
  Galileo, 8
  INS aiding, 475
Schmidt, Jeff, xxxi
Schmidt, Stanley F., 413, 431
Schmidt-Kalman filter, 413
Schuler
  oscillation, 103, 458
  period, 458
Schuler, Maximilian, 457
Scoresby test, 103
SDR, 242–243
Selective availability, 5, 379
Self-alignment (INS), 58
Self-calibration (INS), 79–80
Semi-major axis, 507, 509
Sensor, 351
  calibration, 56, 63–66
  compensation, 63–64
  error model, 59
  inertial, 380
  input axis, 56
  misalignments, 63–65
  multi-axis, 56
  noise, 360
    common-mode, 361
    distribution matrix, 361
  nonlinearity, 384
SFAP, 180, 185
SFIR, 17
Shock isolators, 58, 78
Sidereal rotation, 70
Signal structure
  Compass/BD, 146
  Galileo, 145–146
  GLONASS, 142–144
  GPS, 116–141
  QZSS, 147
Simulation
  figure-8 track, 408
  performance, 412
Singular value decomposition, 386
SK filter, 414

Slater, John, 21
SLBM, 29
SOL, 7
Specific force, 27, 55, 89
Spreading code, 126
Spread spectrum, 206
SPS, 109
Sputnik, 474
Square root filter, 418–421
  Bierman-Thornton, 419–421
  Carlson, 419–421
  Morf-Kaiath, 419
  Potter, 418–420
SSBN, 474
Stable element, 56, 58, 78
Stable platform, 56, 78
STAP, 180
Star tracker, 57, 67, 85
State transition matrix, 373
State variables, 351–352
State vector, 352
STM, 373
Stochastic, 365
  calculus, 365
  differential equation, 365
  systems, 365
Strapdown (INS), 28, 57, 78, 82
  advantages, 82
  disadvantages, 82
SVD, 386
Systems analysis (INS), 430

Temporal update, 353
Tilts (INS), 434
TIMATION, 474
Time of arrival, 214
Time of transmission, 225–226
TLM, 115
TOW, 115
Tracking filter, 397
TRANSIT, 474
Transverse curvature, 514
True anomaly, 507
TTFF, 204, 236

URE, 117
USAF 621B, 474

Vehicle attitude, 518
Vehicle tracking filter, 397
Vernal equinox, 503
Vertical channel, 68, 90, 452
  altimeter aiding, 454, 456
  stabilization, 90
Vibration isolators, 58, 78

Wagner, Jörg, 10
Wavelength, 226–227
Week number (WN), 115
Weiffenbach, George. C., 474
Weill, Lawrence R., xxxi
WGS, 70
WGS84 geoid, 513
Whitehead, Robert, 15
White noise, 353, 365
Wide-lane, 227
Wiener process, 378
Wrigley, Walter, 10, 34

Yaw angle, 519
Y-code, 130

Z-count, 114
ZLG, 23