

Orbit Determination Tool Kit

Theory & Algorithms¹

James R. Wright, *et al.*

November 1, 2013

¹© Analytical Graphics, Inc., 2002 through 2014. All Rights Reserved.

Contents

0.1	Preface	xx
0.2	Acknowledgements	xx
0.3	US Patents	xx
1	Introduction	1
1.1	Orbit Determination Classes	1
1.1.1	IOD Methods	2
1.1.2	LS Methods	2
1.1.3	SP Methods	2
1.1.4	"Batch Filter" (Sequential LS) Methods	3
1.1.5	Optimal Orbit Determination	3
1.2	Notation	4
1.2.1	Coordinate Frames	4
1.2.2	Spacecraft Vectors and Components	7
1.2.3	Numerical Orbit Propagation	8
1.3	Mathematical Operators for SP Methods	9
1.3.1	Subscript Notation	9
1.3.2	Nonlinear Operators	10
1.3.3	Linear Operators	10
1.4	Document Partition	11
I	Optimal Orbit Determination	13
2	Optimal Orbit Determination (OOD)	15
2.1	Definitions	15
2.1.1	State Estimate Reference for Linearization of SP Methods	15
2.1.2	Local Linearization	16
2.1.3	Global Linearization	16
2.1.4	Observability	16
2.1.5	Completeness	16
2.1.6	Optimal Orbit Determination	16
2.1.7	Discussion	17

3	Fundamental Theorem of Estimation	19
3.1	Gaussian Probability Density Function	19
3.1.1	Sample Data	19
3.1.2	Multidimensional Gaussian Random Variable	19
3.2	Gaussian Cumulative Distribution Function	20
3.2.1	Central Limit Theorem	21
3.2.2	Multidimensional Gaussian Random Variable	22
3.3	Short Version of Fundamental Theorem	22
3.4	Complete Version of Fundamental Theorem	22
3.5	Comparison to Least Squares	22
4	Kalman's Approach	23
4.1	Linearization	23
4.2	State Estimate Error Model	24
4.3	Integral Equation	24
4.3.1	Gauss-Markov Sequence	25
4.4	Time Update Algorithm	25
4.4.1	State Error Covariance for Filter Time Update	25
4.5	Measurement Update Algorithm	26
5	Optimal Sequential Filter	29
5.1	State Estimate Error Model	29
5.2	Integral Equation	30
5.3	State Error Covariance for Filter Time Update	30
5.4	Approximate Error-Model Solutions	31
5.4.1	Gravity Solution	31
5.4.2	Air-Drag Solution	31
5.4.3	Solar Pressure Solution	32
5.5	Filter Time Update Algorithm	32
5.6	Filter Measurement Update	32
5.6.1	Measurement Editing	32
5.6.2	Supplementary Editor	33
5.6.3	Filter Initialization	33
5.6.4	Filter Divergence	33
6	Fixed Interval Sequential Smoother	35
6.1	Smoother Initialization	35
6.2	Notation for Smoother Nonlinear State Transition	36
6.3	Smoother Sequential Equations	36
6.3.1	Transition Smoothed State Estimate Backwards	36
6.3.2	Incorporate Filter Estimate and Covariance at Time t_k	36
6.3.3	Prepare to Calculate Smoother Covariance	36
6.3.4	Smoother Covariance	36
6.3.5	Notes	37
6.3.6	Example for $k = L - 1$	37
6.4	Filter-Smoother Consistency Test	37

6.4.1	Test	38
6.5	Smoother Matrix Inversions	38
7	Variable-Lag Sequential Smoother	39
7.1	Introduction	39
7.2	General Application	40
7.2.1	ODTK LEO Simulation	41
7.3	Impulsive Maneuvers	41
7.4	Required Properties	42
7.5	Properties Available	43
7.5.1	Linear	43
7.5.2	Extension	44
7.5.3	Nonlinear	44
7.6	Kalman Filter	45
7.6.1	Time Update	45
7.6.2	Measurement Update	46
7.7	Carlton-Rauch Fixed-Epoch Smoother	47
7.7.1	FES Initialization	47
7.7.2	Measurement at $t_j = t_{k+1}$	48
7.7.3	Measurements at $t_j = t_{k+1}, t_{k+2}, \dots$	49
7.8	Frazer Fixed-Epoch Smoother	49
7.8.1	FES Initialization	49
7.8.2	Measurement at $t_j = t_{k+1}$	49
8	State-Space Changes	51
8.1	Problem Definition	51
8.1.1	Structure of VLS Algorithms	52
8.2	Handling Discontinuities by the Filter	52
8.2.1	State Contraction	52
8.2.2	State Expansion	54
8.2.3	Simultaneous Contraction and Expansion	54
8.2.4	Stability Properties of the Covariance Contraction	54
8.3	Smoothing Algorithms	56
8.3.1	Fixed-Interval Smoothing (RTS)	56
8.3.2	Fixed-Point Smoothing (Carlton-Rauch)	57
8.3.3	Fixed-Point Smoother (Meditch, Anderson-Moore)	57
8.3.4	McReynolds Multi-point Smoother	60
8.4	Handling Filter Discontinuities with Smoothers	61
8.4.1	Fixed-Interval Smoothing with Discontinuities	61
8.4.2	Fixed-Point Smoothing (Carlton-Rauch) with Discontinuities	63
8.4.3	Fixed-Point Smoothing (Fraser) with Discontinuities	64
8.4.4	Multi-Point Smoothing with Discontinuities	65

9	Relative Orbit Errors	67
9.1	Simultaneous Orbit Determination	67
9.2	Orbit Difference Error Covariance	68
10	Keplerian Variables	69
11	Equinocial Variables	71
12	Time Grids	73
II	Stochastic Sequences	75
13	Stochastic Sequences for OOD	77
13.1	A Scalar Exponential Gauss-Markov Sequence	77
13.1.1	Deterministic Transitivity with Time	78
13.1.2	Stationary Variance	79
13.1.3	Propagation Time Extrema	80
13.1.4	Input Control	80
13.1.5	Estimation	81
13.1.6	Stationarity	82
13.2	The Vasicek Stochastic Sequence	82
13.2.1	Transition Matrix $\bar{\Phi}$	84
13.2.2	Units	85
13.2.3	Volatility	85
13.2.4	Drift	85
13.2.5	Mean	85
13.2.6	Second Moments	85
13.2.7	Summary	87
13.2.8	Example	87
13.2.9	Sequential Estimation	88
13.3	Other Stochastic Sequences	92
13.3.1	Brownian Motion	92
13.3.2	White Noise	94
13.3.3	Clocks	96
III	Accelerations	97
14	Accelerations	99
14.1	Two Body Acceleration	99
14.2	Total Acceleration	99

15 Earth Gravity	101
15.1 Geopotential	101
15.2 Legendre Functions	102
15.2.1 Legendre Polynomials of First Kind	102
15.2.2 Associated Legendre Functions of First Kind	103
15.3 Auto-Covariance Function on a Sphere	103
15.3.1 From Angle to Time	105
15.3.2 Covariance Function $R(0)$	105
15.3.3 Auto-correlation Function	105
15.4 Acceleration Errors on a Sphere	106
15.4.1 Computational Problem	107
15.4.2 Solution to Computational Problem	108
15.4.3 The Double Integral	108
15.4.4 Markov Property	108
15.4.5 Inertial Frame	109
15.4.6 Stochastic Characterization	110
15.4.7 Implementation	110
15.5 Perigee-Apogee Weighting	110
15.5.1 Weighting Functions	111
15.6 New Method	112
15.6.1 Splines	112
15.6.2 Sample Covariance Validation	112
15.6.3 GRACE	112
15.6.4 EGM96	113
15.7 Simulate Gravity Coefficient Errors	113
15.7.1 Covariance Matrix Representations	113
15.7.2 Simulation of \mathbf{x}	114
15.7.3 A Useful Example	115
16 Lunar Gravity	117
16.1 Lunar Prospector	117
17 Atmospheric Drag and Lift	119
17.1 King-Hele Unit Vector \mathbf{K}	119
17.2 Orthonormal Vector Basis $[\mathbf{a}]$	120
17.3 Drag-Lift Acceleration	121
17.4 Air-Drag Acceleration	121
17.4.1 Drag Acceleration	121
17.5 Ballistic Coefficient Errors	122
17.6 Atmospheric Density Errors	122
17.6.1 Relative Error Base-Line Model on Air-Density	123
17.6.2 Gauss-Markov Sequence on Air-Density Relative Error	124
17.6.3 Propagation of State Estimate	124
17.6.4 State Estimate Error	125
17.6.5 Process Noise Covariance	125
17.6.6 Transform From Perigee Height to Current Height	127

17.6.7 Orbit Error Covariance	128
18 Solar Photon Pressure	131
18.1 Coordinate Frame for Solar Pressure	131
18.2 Solar Pressure Acceleration	132
18.2.1 Notation	133
18.2.2 Conservation of Linear Momentum	133
18.2.3 Spherical Surface Diffuse Reflection	134
18.2.4 Spherical Surface Perfect Absorption	134
18.3 Eclipse Modeling	134
18.3.1 Selection	134
18.3.2 Baker's Dual-Cone Eclipse Model	136
18.3.3 Earth Radius	137
18.4 Stochastic Solar Pressure Error Model	137
19 GPS Solar Pressure Models	139
19.1 GSPM.04a	139
19.2 GSPM.04ae	139
19.3 AeroT20	139
19.4 AeroT30	140
19.5 Estimation	140
20 Spacecraft Thrusting	141
20.1 Impulsive Maneuver Model	141
20.1.1 Trajectory	141
20.1.2 Trajectory Error Covariance	142
20.1.3 Time of Centroid t_C	143
20.2 Impulsive Maneuver Error Covariance	144
20.2.1 Notation	144
20.2.2 Time Relations	144
20.2.3 Covariance	145
20.3 Impulsive Maneuver Covariance	146
20.3.1 Discussion	147
20.4 Finite Maneuver Model	148
20.4.1 Kinematics	148
20.4.2 Dynamics	149
20.4.3 Stochastic Sequences	149
20.4.4 Estimation	150
IV Spacecraft Attitude	151
21 Attitude Modeling	153
21.1 Antenna Phase Center Estimation	153

V	State Error Transition	155
22	Transitive Partial Derivatives	157
22.1	State Error Transition Function	157
22.2	Position & Velocity Partial	157
22.2.1	Two-Body Transition Matrix	158
22.2.2	Variational Equations Transition Matrix	158
22.3	Air-Drag Partial	161
22.4	Solar Pressure Partial	162
VI	Ground Location Estimation	163
VII	Measurements	167
23	Tracking Station Kinematics	169
23.1	Introduction	169
23.2	The Earth-Centered Unit Sphere	170
23.3	Ellipse in Plane of \mathbf{f}_1 and \mathbf{f}_3	172
23.3.1	Normal Vector $\mathbf{n}(p^f)$	173
23.3.2	Geodetic Latitude	173
23.3.3	Normal Vector $\mathbf{n}(\varphi)$	174
23.3.4	Station Height Above Ellipse	174
23.3.5	\mathbf{R}	175
23.3.6	Geocentric vs Geodetic Latitude	175
23.3.7	Topocentric Geodetic Vector Basis $[\mathbf{N}]$	178
23.3.8	Topocentric Geocentric Vector Basis $[\mathbf{M}]$	180
23.4	Earth Fixed Vector Basis $[\mathbf{e}]$	182
23.4.1	From $[\mathbf{f}]$ to $[\mathbf{e}]$	182
23.4.2	The Ellipsoid	182
23.4.3	Station Vector	182
23.4.4	Normal Vector $\mathbf{n}(\varphi)$	183
23.4.5	\mathbf{R}	183
24	Angles	185
24.1	Unit Range Vector	185
24.2	Azimuth and Elevation	185
24.2.1	Construct \mathbf{L} from Azimuth and Elevation	186
24.2.2	Construct Azimuth and Elevation from \mathbf{L}	187
24.2.3	Partial Derivatives of Azimuth and Elevation	187
24.2.4	Construct L^e from Azimuth and Elevation	189
24.2.5	Construct Azimuth and Elevation from L^e	190
24.3	Direction Cosines	190
24.3.1	Construct Direction Cosines L_1^F and L_2^F from \mathbf{L}	192
24.3.2	Partial Derivatives of Direction Cosines L_1^F and L_2^F	192

24.4	<i>X/Y</i> Angles	193
24.4.1	<i>X/Y</i> East-West Reference Frame	193
24.4.2	<i>X/Y</i> North-South Reference Frame	194
24.4.3	<i>X/Y Z</i> Reference Frame	194
24.4.4	<i>X</i> and <i>Y</i> Angles	194
24.4.5	Partial Derivatives of <i>X</i> and <i>Y</i> Angles	195
24.5	Ground Based Tracker	195
24.5.1	Partials	195
24.6	Space Based Tracker	197
24.7	Earth Ellipsoid Values	197
24.7.1	NIMA/NASA EGM96	197
24.7.2	On the EGM96 Ellipsoid	198
25	Range	199
25.1	Classical Two-Way Range	201
25.1.1	Notation	201
25.1.2	Definitions	201
25.1.3	Special Relativity	202
25.1.4	Calculation	202
25.1.5	Complete Representation	203
25.2	Satellite to Satellite Two-Way Range	203
25.3	Bi-Static Range (One-Way)	203
26	Doppler	205
26.1	Classical Two-Way Doppler	206
26.1.1	Notation	206
26.1.2	Definitions	206
26.1.3	Expressions for Uplink Range-Rate	207
26.1.4	Expressions for Downlink Range-Rate	208
26.1.5	Two-Way Range-Rate	209
26.1.6	Two-Way Doppler Frequency Equation	210
26.1.7	Two-Way Phase Count Equation	211
26.1.8	Observed Frequency	211
26.1.9	Measurement Names	212
26.2	SCF ARTS Doppler	212
26.2.1	Complete Representation	213
26.2.2	Partial Derivatives	213
26.3	NASA STDN Doppler	214
26.3.1	Observed Range Rate	214
26.3.2	Doppler Representation from the Observed Range Rate Equation	215

27 Troposphere	217
27.1 Troposphere Range	217
27.1.1 Zenith Component	217
27.1.2 Mapping Function	220
27.2 Partial Pressure Measurements	223
27.3 Atmospheric Thermodynamics	224
27.3.1 State Variables	224
27.3.2 Ideal Gas Law	225
27.3.3 Relative Humidity and Partial Pressures	225
27.4 Troposphere Range Error	226
27.4.1 Propagation Variance for Relative Troposphere Range Error	227
27.4.2 Sequential Estimation	228
27.5 SCF 'RC Model	228
27.5.1 Tropospheric Range Model ('RC)	228
27.5.2 Doppler	229
28 Ionosphere	231
28.1 Range	231
28.2 Doppler	232
28.2.1 SCF ARTS Doppler (Range-Rate)	232
29 TDOA	233
29.1 Ground-based TDOA (Ground to Spacecraft to Ground)	233
29.1.1 Ground-based TDOA light time algorithm	233
29.1.2 Complete Representation	235
29.2 Ground TDOA (Spacecraft to Ground)	235
29.2.1 Ground TDOA light time algorithm	235
29.2.2 Complete Representation	236
29.3 Ground-based single differenced TDOA (Ground to Spacecraft to Ground)	236
29.4 Space-based TDOA (Ground to Spacecraft)	236
29.4.1 Space-based TDOA light time algorithm	237
29.4.2 Complete Representation	237
30 FDOA	239
30.1 Ground-based FDOA (Ground to Spacecraft to Ground)	239
30.1.1 Doppler shift	239
30.1.2 Frequency of arrival	240
30.1.3 Complete Representation	240
30.2 Ground FDOA (Spacecraft to Ground)	241
30.2.1 Frequency of arrival	241
30.2.2 Complete Representation	241
30.3 Ground-based single differenced FDOA (Ground to Spacecraft to Ground)	241
30.4 Space-based FDOA (Ground to Spacecraft)	242

30.4.1	Frequency of arrival	242
30.4.2	Complete Representation	242
31	TDOA Dot	243
31.1	Ground-based TDOA Dot (Ground to Spacecraft to Ground) . .	243
31.1.1	Range rate	243
31.1.2	Rate of change of time of arrival	244
31.1.3	Complete Representation	244
31.2	Space-based TDOA Dot (Ground to Spacecraft)	244
31.2.1	Rate of change of time of arrival	245
31.2.2	Complete Representation	245
32	FDOA Dot	247
32.1	Ground FDOA Dot	247
32.1.1	Frequency of arrival rate	247
32.1.2	Complete Representation	248
33	DSN Range and Total Count Phase	249
33.1	Sequential Range	249
33.2	Total Count Phase	249
33.3	Doppler	250
33.4	Antenna Corrections	250
33.5	Media Corrections	250
33.6	Solar Corona Model	250
34	Clock Modeling	251
34.1	Introduction	251
34.2	Time	252
34.2.1	Variable t	252
34.2.2	Constant τ	252
34.3	ODTK Simulator Clock Time Update	252
34.4	ODTK Filter Clock Time Update	253
34.4.1	State	253
34.4.2	Covariance	253
34.5	Allan Variance	253
34.5.1	Covariance on Clock Phase	254
34.5.2	Simulations	255
34.5.3	Filter Time Update for Clock	258
34.5.4	State Estimate Parameters	258
34.6	Zucca-Tavella Clock Model	259
VIII	Least Squares (LS)	261
35	LS Inputs	263
35.1	Input Values	263

35.2	Initial Calculated Values	264
35.2.1	Measurement Residuals	264
35.2.2	Partial Derivatives	265
36	Least Squares Solutions	267
36.1	LS Normal Equation and Solution	267
36.1.1	Linearization	267
36.2	Remove the Squaring Operation	268
36.3	Solution by Triangularization of B	269
36.3.1	Solution Overview	269
37	Least Squares Inadequacies	271
37.1	LS Measurement Residuals	271
37.2	Incomplete LS Model	271
37.3	Least Squares State Error Covariance	271
37.4	Batch Simultaneity	272
37.5	Schmidt's Analysis of Least Squares	272
37.5.1	A Practical Experiment	272
37.6	Gibb's Effect	273
38	Tracking Data Editing	275
38.1	Tracking Data Editor Identification	276
38.2	Minimum Number of Data Sets Editor	277
38.3	Minimum Time Span Editor	277
38.4	Tracking Loop Flag Editor	277
38.5	Gross Raw Data Editor	277
38.6	Minimum Elevation Editor	278
38.7	Sliding Raw Data Polynomial Editor	278
38.8	Kepler Element Editor	278
38.8.1	Kepler Element Bounds Criterion	279
38.8.2	$m = M = 3$	281
38.8.3	$m = M = 4$	281
38.8.4	$m = M = 5$	281
38.8.5	$m \geq 6$	282
38.8.6	Kepler Element Statistics Criterion	282
38.9	Second Difference Editor Using IOD	283
38.10	Convergence Criteria	283
38.10.1	Residual RMS Convergence Criterion Using IOD	283
38.10.2	Residual RMS Convergence Criterion After First Correction	284
38.11	Residual Editor	285
38.12	Least Squares Solution	285
38.12.1	Exceed Maximum Iteration Count Editor	285
38.13	Manual Editor	285
38.14	Select A New Station Pass	285
38.15	Measurement Biases	285

IX	Initial Orbit Determination	287
39	IOD Methods	289
39.1	Common Modeling Limitations	289
39.1.1	Two-Body Dynamics	289
39.1.2	Measurement Outliers	289
39.2	Common Equations for IOD	290
39.2.1	Equation of Motion	290
39.2.2	Triangle Geometry	290
40	Herrick-Gibbs	291
40.1	Position Vectors	291
40.1.1	Taylor's Series	291
41	Gooding	295
41.1	Description	295
41.1.1	Guess Range Values	295
41.1.2	Lambert	295
41.1.3	Gooding	296
41.2	A Priori Orbit Information	297
41.2.1	Near-Circular Orbits	297
41.2.2	High Eccentricity Orbits	298
41.3	White Noise	298
41.4	Tropospheric Effects	298
41.5	Inspection of Kepler Orbit Element Values	298
41.6	Multiple Solutions from Each Set of Distinct Measurement Sets	298
41.7	Least Squares	299
X	Tracking and Data Relay Satellite System (TDRSS)	301
42	TDRSS Range and Doppler	303
42.1	TDRSS Range Vectors	303
42.2	Range Sum	304
42.2.1	Range Elements	304
42.2.2	Range Sum Definition	305
42.2.3	Range Sum Time Tag	305
42.2.4	Construction of the Range Sum Representation	305
42.3	Range Sum Partial Derivatives	306
42.3.1	Eliminate Light-Time	306
42.3.2	Simplified Expressions	307
42.3.3	Differentials	307
42.3.4	LEO Satellite Range Sum Partial	307
42.3.5	TDRS Range Sum Partial	309
42.4	Doppler Measurements	310
42.5	DopplerPartials	311

42.5.1	LEO Satellite Doppler Partial	311
42.5.2	TDRS Doppler Partial	312
43	One-Way Return-Link Doppler	313
XI	Global Positioning System (GPS) Low Earth Orbit (LEO) Receivers	315
44	Pseudo-Range Filtering	317
44.1	Introduction	317
44.1.1	Pseudo-Range Option	317
44.2	USER GPS Pseudo-Range	318
44.2.1	Pseudo-Range	318
44.2.2	Range Measurement Time Notation	318
44.2.3	Time Differences	318
44.2.4	Receiver Clock Errors	318
44.2.5	Receiver Proper Time less Receiver Coordinate Time (User Spacecraft)	319
44.2.6	Coordinate Time Difference	320
44.2.7	Transmitter Coordinate Time less Transmitter Proper Time (NAVSTAR)	320
44.2.8	Transmitter Clock Error	321
44.3	Schwarzschild Metric	322
44.3.1	Proper Time vs Coordinate Time	323
44.3.2	Time Integrals	323
44.3.3	JPL Model	325
44.4	Group Delay Differential	325
44.5	Remove Ionospheric Range	325
44.5.1	Proof	326
44.5.2	Phase Count	326
44.6	GPS Range First-Differences	327
44.6.1	Notation	327
44.6.2	C/A Code Range Representation	327
44.6.3	First Difference	327
44.7	Partial Derivatives for HANU	328
44.7.1	ICD-GPS-208 18 April 1983	328
44.7.2	Partials	328
44.7.3	Map HANU Covariance P to GPS Measurement Covariance P_y	329
44.7.4	Initial Differential Equations for Variations in Range and Range-Rate	330
44.7.5	Position and Velocity Variations in Inertial Component Matrices	332
44.7.6	Partials	332

45 GPS Carrier Phase Count	335
45.1 Introduction	335
45.1.1 BlackJack GPS Receiver	336
45.2 Implementation	336
45.2.1 Estimation of Clock Parameters	336
45.2.2 L_1 Phase Count Measurements	337
45.2.3 L_1, L_2 Phase Count Measurements	337
45.2.4 Partial Derivatives	337
45.3 Notation	337
45.3.1 Time	337
45.3.2 Oscillator and Clock	338
45.3.3 Generalized Integral	338
45.3.4 Doppler Phase Count	338
45.3.5 Two Types of Phase Count Error	339
45.4 Receiver Doppler Algorithm	340
45.4.1 NAVSTAR Frequencies	340
45.4.2 Doppler Frequency	340
45.4.3 Define Phase Count	341
45.4.4 Doppler Shift	342
45.4.5 Phase Count Representation	343
45.4.6 Approximations for χ_{ij}	344
45.4.7 USER Orbit Partial Derivatives	345
45.5 Ionosphere	346
45.5.1 Physics	346
45.5.2 Total Electron Content	347
45.5.3 Validation	349
45.6 Final Forms for Phase Count	349
45.6.1 Alternate Representation	350
45.7 Ionosphere Removal	351
45.7.1 Ionosphere-Free White Noise Variance for \bar{N}_j^{Ei} (Cycles)	353
45.7.2 Ionosphere-Free White Noise Variance for $\Delta\rho$ (cm)	355
45.8 First Differences on N_j^{Ei}	358
45.8.1 Remove USER Clock Phase Perturbations	358
45.8.2 Root-Variance Mappings	359
45.9 Example	360
45.10 Partials and Covariance	360
45.10.1 Single Frequency	360
45.10.2 Two-Frequency Ionosphere Removal	361
45.10.3 First Differences on N_j^{Ei}	362
45.11 Receiver Clock Error Model	363
45.11.1 Covariance on Clock Phase	363
45.11.2 Simulations	365
45.11.3 Filter Time Update for Receiver Clock	366
45.11.4 State Estimate Parameters	367

46 GPS Navigation Solution	369
46.1 Introduction	369
46.2 Algorithm	369
XII Global Positioning System (GPS) Ground Receivers	
373	
47 GPS Composite Clock	375
47.1 GOMA-E	375
47.2 Tutorial	375
47.2.1 Simulation Narrative	375
47.2.2 Similarity in Clock Parameter Variations	376
47.2.3 Optimal Estimation of the UVCC	377
47.2.4 Real World Application	377
XIII Appendices	383
A. The Least Squares Quadratic	385
B. Sequential Least Squares	387
C. Ad-Hoc Batch Filter	391
D. Kalman's Model Equation	395
.0.5 Measurements at $t_j = t_{k+1}, t_{k+2}, \dots$	395

List of Figures

3.1	Gaussian Density Function $f(\mathbf{x})$	20
3.2	Gaussian Distribution Function $F(\mathbf{x})$	21
7.1	Variable Lag Smoother with EKF and FES	40
7.2	Estimation of Velocity at Fixed Epochs with EKF and FES	41
13.1	Exponential Transition-Correlation Function and Half-Life Function	88
13.2	24 Unbiased Vasicek Sequences	89
13.3	24 Biased Vasicek Sequences	89
17.1	Sigma for Relative Error in Air Density	123
18.1	Dual-Cone Eclipse Geometry	135
18.2	Baker's Solar Obscura	136
20.1	Unit Thrust Referred to Maneuver Frame	148
23.1	Station Longitude Rotation	171
23.2	Geocentric Station Vector Basis	172
23.3	Station Position Vector	175
23.4	Geocentric vs Geodetic	176
23.5	Vector Basis for Angles Measurements	179
23.6	Latitude Rotation	180
23.7	Latitude Difference Rotation	181
24.1	Azimuth-Elevation Description	186
24.2	Direction Cosines Description	191
24.3	Unit Range Vector Right Ascension & Declination	196
34.1	Simulated Allan Variance Diagram	257
37.1	Position Error vs Least Squares Fit Span	272
42.1	Four TDRSS Range Vectors	304

44.1 Earth-USER-NAVSTAR Vector Triangle	330
47.1 Sim Phase Dev (units: 1×10^{-7} s) vs Time (0 - 8) d	378
47.2 Sim & Est Phase Dev (units: 1×10^{-7} s) vs Time (0 - 8) d	379
47.3 4 Similar Phase Var (units: 1×10^{-7} s) vs Time (0 - 8) d	379
47.4 4 Similar Phase Var (units: 1×10^{-8} s) vs Time (0 - 3) d	380
47.5 S1 Phase Var & Cov Bounds (units: 1×10^{-6} s) vs Time (0 - 8) d	380
47.6 S1 Phase Var Reduced Cov (units: 1×10^{-7} s) vs Time (0 - 8) d	381
47.7 S1 Phase Var Reduced Cov (units: 1×10^{-9} s) vs Time (0 - 8) d	381
47.8 4 Similar Phase Var no IC (units: 1×10^{-8} s) vs Time (0 - 8) d	382

0.1 Preface

Analytical Graphics presents here the mathematical specifications for its algorithms and software used to define new capabilities for orbit determination. These capabilities are referred to as Orbit Determination Tool Kit (ODTK¹).

The initial purpose for this document was to enable the author to proceed with the construction of new prototype software for the AGI ODTK methods of orbit determination. Recording specific algorithm definitions, and their relations to appropriate hardware dynamics and spacecraft trajectory physics, enables and supports algorithm development, review, validation, modification, and communication.

This is a living document. Its secondary purpose is to provide relevant mathematical specifications for ODTK users as the algorithms and software are developed.

This document is constructed and modified using the Standard LaTeX Book shell for Scientific Workplace 5.5 by MacKichan Software.

0.2 Acknowledgements

I am especially appreciative of the contributions by Dr. James Woodburn to ODTK, and to this document. I thank Dick Hujsak and William Chuba for useful suggestions in development of the first differencing algorithm for GPS Doppler carrier phase count measurements.

0.3 US Patents

Patent No. US 6,708,116, *Method and Apparatus for Orbit Determination*, was granted 16 March 2004.

Patent No. US 7,050,002, *GPS Carrier Phase Measurement Representation and Method of Use*, was granted 23 May 2006.

Patent No. US 8,275,498 *Nonlinear Variable Lag Smoother*, was granted 30 April 2013.

¹ODTK was previously known as STK/OD.

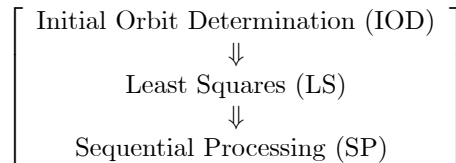
Chapter 1

Introduction

Orbit determination refers to the estimation of orbits of small objects relative to primary celestial bodies, given applicable measurements. All useful orbit determination methods produce orbit estimates that have errors.

1.1 Orbit Determination Classes

Orbit determination methods are partitioned by three classes: Initial orbit determination (IOD), batch least squares differential corrections (LS), and sequential processing (SP). An orbit estimate output from IOD can be used as orbit estimate input to initialize LS, and an orbit estimate output from LS can be used as orbit estimate input to initialize SP:



When there exists an orbit estimate sufficient to initialize SP, then IOD and LS are not required.

	prior orbit est	prior error cov	size of est errors	execution speed
IOD	No	No	Huge	Very Fast
LS	Yes	No	Small	Fast
SP	Yes	Yes	Small	Fast Enough

Table 1.1: Compare Classes of OD Methods

1.1.1 IOD Methods

IOD methods input tracking measurements with tracking platform locations, and output spacecraft orbit estimates. *No* prior orbit estimate is required, and when used, it is assumed that no prior orbit estimate exists; however, orbit estimation errors are usually very large. IOD methods are nonlinear and six-dimensional. Measurement editing is not possible during IOD calculations because no prior orbit estimate exists. IOD methods were derived by various authors: LaPlace, Poincaré, Gauss, Lagrange, Lambert, Gibbs, Herrick, Williams, Stumpp, Lancaster, Blanchard, Gooding, and Smith. The orbit determination process is frequently begun, or restarted, with IOD.

1.1.2 LS Methods

The LS method requires the input tracking measurements with tracking platform locations and a prior orbit estimate. The LS method outputs a refined orbit estimate. Estimation errors are small when compared to IOD outputs. LS methods consist of a sequence of linear LS (differential) corrections where sequence convergence is defined a minimization of the (weighted) sum of squares of measurement residuals. The sum of squares is represented by Eqs. 36.1, 36.5, and 36.7. Minimization is characterized by Eqs. 36.8 and 36.9. LS methods produce refined orbit estimates in a batch mode, together with error-covariance matrices that are optimistic; *i.e.*, LS orbit element error variance estimates are typically too small, relative to truth, by at least an order of magnitude. LS methods require an inversion of an $n \times n$ LS information matrix for each LS iteration, where n is state estimate size. This inversion is problematic when the information matrix is ill-conditioned, which is frequently the case in orbit determination applications of LS. Operationally, LS may be the only method used, or, it may be used to initialize SP. The existence of a prior orbit estimate enables operational measurement residual editing, but LS methods frequently require inspection and manual measurement editing by human intervention. LS algorithms therefore require elaborate software mechanisms for measurement editing. Gauss reputedly developed LS as early as 1795 and used it to determine the orbit of Ceres in 1801, although he did not publish his account of it until the method had already been independently developed and published by Legendre. [26].

1.1.3 SP Methods

SP methods require the input of tracking measurements with tracking platform locations, priori state estimates (including an orbit estimate), and a prior state-error covariance matrix. SP methods output refined state estimates in a sequential mode. SP filter methods process forward in time, repeating a recurring, two-step sequence of *filter time-update* of the state estimate and *filter measurement-update* of the state estimate. The *filter time-update* propagates the state estimate forward, and the *filter measurement-update* incorporates the

next measurement.

The recursive sequence requires an important interval of *filter initialization*. Often the state-error covariance is unrealistic during the filter-initialization time interval because the error covariance matrix associated with the *a priori* orbit estimate is unknown. A significant number of observations must be processed to bring the covariance to a realistic scale. Conceptually, the filter has been initialized once the state covariance becomes realistic; in practice, the filter is considered initialized after having processed a prescribed number of observations.

No matrix inversions are performed during filter calculations. SP smoother methods process backward in time, repeating a pattern of state-estimate refinement using filter outputs and backwards transition. Matrix inversions are required by the smoother algorithm. Computations for both filter and smoother are dominated most significantly by numerical orbit propagations. The search for non-stationary sequential processing was begun by Kalman [49], Bucy [10], Rauch [97], Meditch [77], and others.

1.1.4 "Batch Filter" (Sequential LS) Methods

The batch filter (Appendix C), BF, is a hybrid algorithm that combines properties of LS with properties of SP. The BF is suggested by inspection of the sequential form of batch least squares (Appendix B). This sequential form is modified *ad hoc* by inflating the covariance associated with previous data. The BF can be derived by adding a second term *ad hoc* to the LS quadratic form. The first term minimizes the sum of squares of weighted measurement residuals, while the second term binds the new state estimate to the previous state estimate. This two-term quadratic form is referred to as a maximum-likelihood performance function. Matrix inversion is required by the BF, as in LS. The BF was proposed by Swerling [104].

1.1.5 Optimal Orbit Determination

Optimal orbit determination (OOD, defined in the sequel), depends on the use of an optimal sequential-filter design. For current time and future time, the sequential filter output is used directly. For past time, the filter execution is followed by the execution of a fixed interval sequential smoother. The smoothed trajectory estimates are more accurate than filtered trajectory estimates.

Comparison of OOD with LS

It is useful to compare OOD with LS because many systems use LS to provide operational end-products. Table 1.2 presents several operational capabilities that distinguish OOD from LS. In particular, OOD allows for the estimation of time-varying parameters, compared to the estimation of static parameters for LS. In defense of LS, a time-dependent sequence of static parameters may be estimated with LS, thereby approximating a time-variable; however, this is

	OOD	LS
Real-time orbit determination	<i>Yes</i>	No
Force model error structure	<i>Yes</i>	No
The more data the better	<i>Yes</i>	No
Realistic error covariance	<i>Yes</i>	No
Autonomous data editing	<i>Yes</i>	No
Realistic performance assessment	<i>Yes</i>	No
Estimate time-varying meas biases	<i>Yes</i>	No
Estimate time-varying atmospheric density	<i>Yes</i>	No
Estimate time-varying solar pressure	<i>Yes</i>	No
Estimate time-varying transponder bias	<i>Yes</i>	No
Minimize orbit estimate errors	<i>Yes</i>	No
Reinitialize for maneuvers	No	<i>Yes</i>

Table 1.2: OOD vs LS

done at the expense of increasing the LS state-estimate structure for each time-constant. This puts a burden on the analyst to ensure that each time constant is observable (otherwise there will be problems with the LS matrix inversion). OOD handles time-variability autonomously with a single state estimate parameter, and the OOD filter performs no matrix inverse. OOD smoother matrix inverses are invertable .

Consider the estimation of an air drag parameter. The analyst cannot anticipate when atmospheric density will respond to a coronal mass ejection (CME) during solar maximum, nor the time of first measurement following the time of CME impact. Thus, there is no convenient way to appropriately partition sequential time-constants for LS. For OOD, the filter gain on the atmospheric density parameter estimate will be open due to its associated use of K_P and $F_{10.7}$ to drive the state-estimate error covariance. When the first measurement following the time of CME impact is encountered, correction of the atmospheric density parameter estimate will commence at the time tag of that measurement.

1.2 Notation

1.2.1 Coordinate Frames

In the notation that follows, we shall align each vector basis with one of three dextral orthogonal coordinate frames: Inertial, Gaussian, and Frenét. That is to say, each of the three coordinate axes of each coordinate frame shall be colinear with one of the three unit vectors, for each associated vector basis. The origin of each vector basis shall also be coincident with its associated coordinate frame. The positive direction for each coordinate axis is defined by the direction of the

associated unit vector.

Inertial Vector Basis

Let us denote the Earth centered inertial (ECI) orthonormal vector basis for the International Celestial Reference Frame (ICRF) by the "vectrix" [39]:

$$[\mathbf{i}] = \begin{bmatrix} \mathbf{i}_1 \\ \mathbf{i}_2 \\ \mathbf{i}_3 \end{bmatrix} \quad (1.1)$$

where:

$$[\mathbf{i}]^T = (\mathbf{i}_1, \mathbf{i}_2, \mathbf{i}_3), \quad (1.2)$$

where \mathbf{i}_3 is coincident with the conventional pole (Z-direction) of ICRF, where \mathbf{i}_1 and \mathbf{i}_2 are contained in equatorial plane of the ICRF (where \mathbf{i}_1 is in the X-direction), and where $\mathbf{i}_2 = \mathbf{i}_3 \times \mathbf{i}_1$.¹ It is convenient to associate the origin of the inertial vector basis with center of mass of the Earth.

Gaussian Vector Basis

Inertial spacecraft position and velocity vectors are denoted with \mathbf{r} and $\dot{\mathbf{r}}$. The orthonormal Gaussian vector basis:

$$[\mathbf{u}] = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \end{bmatrix}, \quad (1.3)$$

with origin at the spacecraft center of mass, is anchored to the ECI spacecraft position vector \mathbf{r} :

$$\mathbf{u}_1 \equiv \mathbf{U} = \mathbf{r}/r = [\mathbf{i}]^T \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} \quad (1.4)$$

where:

$$r = \sqrt{\mathbf{r} \cdot \mathbf{r}} \quad (1.5)$$

Orbit angular momentum \mathbf{h} :

$$\mathbf{h} = \mathbf{r} \times \dot{\mathbf{r}} \quad (1.6)$$

enables definition of \mathbf{u}_3 :

$$\mathbf{u}_3 \equiv \mathbf{W} = \mathbf{h}/h = [\mathbf{i}]^T \begin{bmatrix} W_1 \\ W_2 \\ W_3 \end{bmatrix} \quad (1.7)$$

¹Prior to ODTK 6, the Earth-centered inertial (ECI) frame was aligned with mean equator and mean equinox at epoch J2000.

where:

$$h = \sqrt{\mathbf{h} \cdot \mathbf{h}} \quad (1.8)$$

The triad is completed with:

$$\mathbf{u}_2 = \mathbf{u}_3 \times \mathbf{u}_1 = [\mathbf{i}]^T \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} \quad (1.9)$$

The rotation 3×3 matrix R_{ui} from $[\mathbf{i}]$ to $[\mathbf{u}]$:

$$[\mathbf{u}] = R_{ui} [\mathbf{i}] \quad (1.10)$$

where:

$$R_{ui} = \begin{bmatrix} U_1 & U_2 & U_3 \\ V_1 & V_2 & V_3 \\ W_1 & W_2 & W_3 \end{bmatrix} \quad (1.11)$$

Note that:

$$[\mathbf{u}]^T = [\mathbf{i}]^T R_{ui}^T \quad (1.12)$$

It is convenient to associate the origin of the Gaussian vector basis with center of mass of a space object.

Frenét Vector Basis

The orthonormal Frenét vector basis:

$$[\mathbf{f}] = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix}, \quad (1.13)$$

with origin at the spacecraft center of mass, is anchored to the ECI spacecraft velocity vector $\dot{\mathbf{r}}$:

$$\mathbf{f}_2 = \dot{\mathbf{r}}/\dot{s} = [\mathbf{i}]^T \begin{bmatrix} f_{21} \\ f_{22} \\ f_{23} \end{bmatrix} \quad (1.14)$$

where:

$$\dot{s} = \sqrt{\dot{\mathbf{r}} \cdot \dot{\mathbf{r}}} \quad (1.15)$$

In common with the Gaussian frame, adopt:

$$\mathbf{f}_3 = \mathbf{u}_3 = [\mathbf{i}]^T \begin{bmatrix} f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} \quad (1.16)$$

The triad is completed with:

$$\mathbf{f}_1 = \mathbf{f}_2 \times \mathbf{f}_3 = [\mathbf{i}]^T \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \end{bmatrix} \quad (1.17)$$

The rotation 3 x 3 matrix R_{fi} from $[\mathbf{i}]$ to $[\mathbf{f}]$:

$$[\mathbf{f}] = R_{fi} [\mathbf{i}] \quad (1.18)$$

where:

$$R_{fi} = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \quad (1.19)$$

Note that:

$$[\mathbf{f}]^T = [\mathbf{i}]^T R_{fi}^T \quad (1.20)$$

It is convenient to associate the origin of the Frenét vector basis with center of mass of a space object.

1.2.2 Spacecraft Vectors and Components

Position, Velocity, and Acceleration

The spacecraft position, velocity, and acceleration vectors are denoted with \mathbf{r} , $\dot{\mathbf{r}}$, and $\ddot{\mathbf{r}}$. Then 3×1 matrices z , \dot{z} , and \ddot{z} for spacecraft position, velocity, and acceleration components can be represented:

$$\mathbf{r} = [\mathbf{i}]^T z \quad (1.21)$$

$$\dot{\mathbf{r}} = [\mathbf{i}]^T \dot{z} \quad (1.22)$$

$$\ddot{\mathbf{r}} = [\mathbf{i}]^T \ddot{z}. \quad (1.23)$$

where:

$$z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} \quad (1.24)$$

$$\dot{z} = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} \quad (1.25)$$

$$\ddot{z} = \begin{bmatrix} \ddot{z}_1 \\ \ddot{z}_2 \\ \ddot{z}_3 \end{bmatrix} \quad (1.26)$$

Position and Velocity Errors

Spacecraft position, velocity, and acceleration vector errors are denoted with $\delta\mathbf{r}$, $\delta\dot{\mathbf{r}}$, and $\delta\ddot{\mathbf{r}}$. Then 3×1 matrices for instantaneous spacecraft position, velocity, and acceleration error components are represented with:

$$\delta\mathbf{r} = [\mathbf{i}]^T \delta z = [\mathbf{u}]^T \delta y \quad (1.27)$$

$$\delta\dot{\mathbf{r}} = [\mathbf{i}]^T \delta \dot{z} = [\mathbf{u}]^T \delta \dot{y} \quad (1.28)$$

$$\delta\ddot{\mathbf{r}} = [\mathbf{i}]^T \delta \ddot{z} = [\mathbf{u}]^T \delta \ddot{y} \quad (1.29)$$

where:

$$\delta z = \begin{bmatrix} \delta z_1 \\ \delta z_2 \\ \delta z_3 \end{bmatrix} = \begin{bmatrix} \delta\mathbf{r} \cdot \mathbf{i}_1 \\ \delta\mathbf{r} \cdot \mathbf{i}_2 \\ \delta\mathbf{r} \cdot \mathbf{i}_3 \end{bmatrix} \quad (1.30)$$

$$\delta y = \begin{bmatrix} \delta y_1 \\ \delta y_2 \\ \delta y_3 \end{bmatrix} = \begin{bmatrix} \delta\mathbf{r} \cdot \mathbf{u}_1 \\ \delta\mathbf{r} \cdot \mathbf{u}_2 \\ \delta\mathbf{r} \cdot \mathbf{u}_3 \end{bmatrix} \quad (1.31)$$

and where the time derivatives follow by putting appropriate dots over $\delta\mathbf{r}, \delta z$, and δy . Insert Eq. 1.12 into Eq. 1.27 to get:

$$\delta z = R_{iu} \delta y \quad (1.32)$$

$$\delta y = R_{ui} \delta z \quad (1.33)$$

where:

$$R_{iu} = R_{ui}^T \quad (1.34)$$

1.2.3 Numerical Orbit Propagation

Notation

Concatenate the 3×1 position and velocity component arrays z and \dot{z} to form the orbit substate:

$$Z = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \end{bmatrix} \quad (1.35)$$

where Z is dynamic: $Z \equiv Z(t)$.

Let φ_z denote the Variation of Parameters in Universal Variables orbit propagator. Given an initial time t_0 , a final time t_f , and an acceleration model $u(Z(t), t)$, then φ_z propagates $Z(t_0)$ from t_0 to t_f using $u(Z(\tau), \tau)$ to obtain $Z(t_f)$. This may be expressed as:

$$Z(t_f) = \varphi_z \{t_f; Z(t_0), t_0, u(Z(\tau), \tau), t_0 \leq \tau \leq t_f\} \quad (1.36)$$

If by φ_z we always imply the use of $u(Z(t), t)$, then we get a very compact notation for Variation of Parameters in Universal Variables:

$$\varphi_z \{t_f; Z(t_0), t_0\} \equiv \varphi_z \{t_f; Z(t_0), t_0, u(Z(\tau), \tau), t_0 \leq \tau \leq t_f\} \quad (1.37)$$

That is:

$$Z(t_f) = \varphi_z \{t_f; Z(t_0), t_0\} \quad (1.38)$$

Ideally, φ_z satisfies the nested transitivity property:

$$Z(t_{k+2}) = \varphi_z \{t_{k+2}; Z(t_k), t_k\} = \varphi_z \{t_{k+2}; \varphi_z \{t_{k+1}; Z(t_k), t_k\}, t_{k+1}\} \quad (1.39)$$

Note that φ_z is a column matrix with 6 elements:

$$\varphi_z = \begin{bmatrix} \varphi_{z1} \\ \varphi_{z2} \\ \varphi_{z3} \\ \vdots \\ \varphi_{z6} \end{bmatrix} \quad (1.40)$$

1.3 Mathematical Operators for SP Methods

1.3.1 Subscript Notation

State Matrices

The state estimate \hat{X} is referenced at two separate times by using the notation [77]:

$$\hat{X}_{j|i} \equiv \hat{X}(t_j|t_i) \quad (1.41)$$

where $i, j \in \{0, 1, 2, \dots\}$. The time t_j to the left of the vertical bar denotes the epoch of \hat{X} which drives the filter time-update function. The time t_i to the right of the bar denotes the time-tag of the last measurement processed to form \hat{X} , which drives the filter measurement-update function. For example, $\hat{X}_{7|6}$ refers to the state estimate at time t_7 , given the last measurement processed at time t_6 , whereas $\hat{X}_{7|7}$ refers to the state estimate at time t_7 , given the last

measurement processed at time t_7 . $\hat{X}_{7|6}$ was obtained by filter time-update of $\hat{X}_{6|6}$ from t_6 to t_7 .

Similar notation is used for the state-estimate correction:

$$\Delta\hat{X}_{j|i} \equiv \Delta\hat{X}(t_j|t_i) \quad (1.42)$$

and the state-estimate-error covariance matrix:

$$P_{j|i} \equiv P(t_j|t_i) \quad (1.43)$$

Measurement Matrices

We denote a measurement taken at time t_j as y_j . We also denote a conditional measurement estimate (representation) at time t_j as $\hat{y}_{j|h}$.

1.3.2 Nonlinear Operators

Nonlinear operators are required in the state estimate time update and the state estimate measurement update for SP methods of orbit determination.

State Propagation

Let φ denote a nonlinear operator that propagates the state estimate $\hat{X}_{i|h}$ from time t_i to time t_j :

$$\hat{X}_{j|h} = \varphi \left\{ t_j; \hat{X}_{i|h}, t_i \right\} \quad (1.44)$$

Measurement Representation

Let $y(\cdot)$ denote a nonlinear operator that calculates the measurement representation $\hat{y}_{j|h}$, given the state estimate $\hat{X}_{j|h}$:

$$\hat{y}_{j|h} = y \left(\hat{X}_{j|h}, t_j \right) \quad (1.45)$$

1.3.3 Linear Operators

State Estimate Error Propagation

Let $\Phi_{j,i} \equiv \Phi(t_j, t_i)$ denote the linear operator that propagates the state error estimate $\Delta\hat{X}_{i|h}$ from time t_i to time t_j

$$\Delta\hat{X}_{j|h} = \Phi_{j,i} \Delta\hat{X}_{i|h} \quad (1.46)$$

where

$$\Phi_{j,i} = \left[\frac{\partial X_j}{\partial X_i} \right]_{\hat{X}_{j|h}} \quad (1.47)$$

and where evaluation derives from $\hat{X}_{j|h}$.

Measurement Residual

Let Δy_j denote the linear operator that defines the measurement residual at time t_j :

$$\Delta y_j = y_j - \hat{y}_{j|h} \quad (1.48)$$

which reflects the discrepancy between the predicted measurement and the actual measurement at time t_j . Some authors refer to Δy_j as "measurement innovation".

Measurement/State Partial Jacobian

Let H_j denote the Jacobian of the measurement with respect to the state partial derivatives at time t_j :

$$H_j = \left[\frac{\partial y_j}{\partial X_j} \right]_{\hat{X}_{j|h}} \quad (1.49)$$

where evaluation derives from $\hat{X}_{j|h}$.

1.4 Document Partition

This document has the following ordered parts:

1. Optimal Orbit Determination
2. Stochastic Sequences
3. Accelerations
4. Spacecraft Attitude
5. State Error Transition
6. Ground Location Estimation
7. Measurements
8. Deep Space Network (DSN)
9. Least Squares (LS)
10. Initial Orbit Determination (IOD)
11. Tracking and Data Relay Satellite System (TDRSS)
12. Global Positioning System (GPS)
13. Appendices

Part I

Optimal Orbit Determination

Chapter 2

Optimal Orbit Determination (OOD)

As previously mentioned, *orbit determination* refers to the estimation of orbits of small objects relative to primary celestial bodies, given applicable measurements. But what is *optimal* orbit determination? The purpose of this chapter is to answer the question.

The adjective *optimal* refers to most desirable, most favorable, or most satisfactory [113]. But most satisfactory to whom? There are choices to make from available orbit determination methods. Should we prefer the fastest methods or the most accurate? Should we prefer sequential methods or batch methods? Should we minimize the size of the weighted measurement residuals or the size of orbit errors? How should we model measurement residuals and orbit errors?

All orbit determination problems are multidimensional and nonlinear, but should we attempt a multidimensional nonlinear solution directly? Or, should we use a linearization method? If so, is there a preferred method for linearization?

2.1 Definitions

2.1.1 State Estimate Reference for Linearization of SP Methods

Evaluation of the measurement representation $\hat{y}_{j|h}$ defined by Eq. 1.45 requires the use of some prior state estimate $\hat{X}_{j|h}$, where $t_j \geq t_h$ and where $\hat{X}_{j|h}$ is the state estimate *reference* for linearization. A similar requirement is associated with Eqs. 1.47 and 1.49.

2.1.2 Local Linearization

Let t_k and $t_{k+1} \geq t_k$ be the times of adjacent ordered measurements y_k and y_{k+1} , for $k \in \{0, 1, 2, \dots\}$. That is, there are no measurements between y_k and y_{k+1} . Then the use of $\hat{X}_{k+1|k}$ as the state estimate *reference* for all linearizations at time t_{k+1} defines *local* linearization at time t_{k+1} .

The use of any state estimate reference other than $\hat{X}_{k+1|k}$ at time t_{k+1} for linearization is a non-local linearization at time t_{k+1} .

2.1.3 Global Linearization

Given the integer variable $k \in \{0, 1, 2, \dots\}$ and given a fixed non-negative integer j , the use of $\hat{X}_{k|j}$ as the state estimate *reference* for linearization at time t_k for each k is known as *global* linearization.

2.1.4 Observability

A particular parameter is *observable* to a particular measurement if and only if the sequential processing of that measurement reduces the estimated error variance on that parameter.

2.1.5 Completeness

The state estimate structure is *complete* if and only if all parameters with unknown observable components are included in the state estimate structure.

2.1.6 Optimal Orbit Determination

By optimal orbit determination, the method used to calculate the state estimate (including the orbit estimate) satisfies the following conditions:

1. Sequential processing (SP) is used to account for force modeling errors and measurement information in the time order in which they are realized.
2. Sherman's Theorem is applied [101],[102],[77],[49]. To summarize, the optimal state estimate correction matrix $\Delta\hat{X}$ is the expectation of the state error matrix ΔX given the measurement residual matrix Δy . That is: $\Delta\hat{X} = E\{\Delta X|\Delta y\}$.
3. Linearizations of state-estimate time transition and state-to-measurement representations are *local* in time, *not global*.
4. The state estimate structure is *complete*.
5. All state-estimate models and state-estimate-error model approximations are derived from the appropriate *physics* of sensors and force modeling.

6. All measurement models and measurement-error model approximations are derived from the appropriate sensor hardware definitions and associated physics, and measurement sensor performance.
7. Necessary conditions for real data include:
 - Measurement residuals approximate Gaussian white noise[77][85]
 - McReynold’s filter-smoother consistency test is satisfied.
8. For simulated data, the state-estimate errors agree with the state-estimate error covariance function.

The first six conditions define standards for optimal algorithm design and for the establishment of a *realistic* state-estimate error covariance function. The last two conditions enable validation: they define realizable test criteria for optimality.

2.1.7 Discussion

Sherman’s Theorem

As a lemma to Sherman’s Theorem [77], one achieves minimum state-estimate error variance (and thus, minimum orbit-estimate error variance). Of the various extremalization criteria available, this one most directly addresses errors in the orbit elements.

Complete State Estimate

Consider any case where the state estimate structure is incomplete. Any component of an observable parameter neglected in the state estimate structure will *alias* into the estimated orbit elements, significantly degrading them. Thus, one needs an appropriate place in the state estimate structure to put every observable effect.

Gaussian White Noise

What is it? In one dimension, one may think of a Gaussian white noise sequence as a sequence of ratios from a random walk sequence R_j , $j \in \{0, 1, 2, \dots\}$. The numerator in each ratio is the difference $(R_{j+1} - R_j)$ in the random walk functional across a specified time interval $[t_j, t_{j+1}]$, and the denominator is the associated time difference $(t_{j+1} - t_j)$. The ratio limit $(t_{j+1} - t_j) \rightarrow 0$ does not exist [19]. Thus, one must always use Gaussian white noise in a granular manner. The Wiener-Levy (random walk) sequence developed in Papoulis [93] provides appropriate results for useful application, and is discussed in the sequel.

Relevance to Observations Gaussian white noise models are appropriate for thermal noise associated with resistance in electronic circuits [99]. Thus, Gaussian white noise is used *directly* for modeling stochastic phenomena in clocks, transmitters, receivers, and sensors. Range and Doppler measurements contain Gaussian white noise.

Gauss used a Gaussian white noise model for orbit determination to represent noise in angles right ascension and declination [26].

Applicability to Linear Systems Gaussian white noise is appropriate, indirectly, as a linear system input to develop a Gaussian stochastic output functional with particular serial correlation properties. This provides a convenient method to represent stochastic modeling errors in some cases. But in other cases, this method is useless; *e.g.*, for acceleration modeling errors that derive from errors in modeling the geopotential.

"Colored" Spacecraft Acceleration Model Errors Acceleration modeling errors that derive from errors in modeling the geopotential, atmospheric density, and solar photon pressure are random and nonstationary, but they are *not* white. Correlation of errors in the time domain are sometimes said to be "colored" noise.

Chapter 3

Fundamental Theorem of Estimation

Measurement-error processes and force-model-error processes are modeled as Gaussian (normal) random functionals for orbit determination. Therefore, all *distributions* considered herein are Gaussian. The Fundamental Theorem of Estimation requires symmetry and left-hand convexity of the distribution function used for estimation. Both requirements are satisfied by Gaussian distributions.

3.1 Gaussian Probability Density Function

The Gaussian density function $f(x)$ on scalar random variable \mathbf{x} is defined by [26]:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (3.1)$$

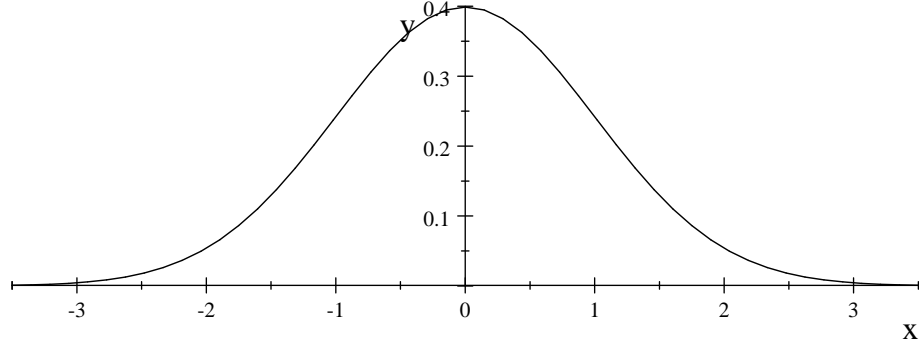
where μ is the mean value of \mathbf{x} , and σ is the root-variance of \mathbf{x} about μ . Figure 3.1 displays $f(x)$ for $\mu = 0$ and $\sigma = 1$.

3.1.1 Sample Data

It can be difficult to see whether a sample is normally distributed - especially if the sample size is small - unless an objective test is employed. For example, a histogram of 100 standard normal deviates cannot be expected to precisely follow Figure 3.1. Statistical hypothesis tests of the normality assumption are useful in such cases [109].

3.1.2 Multidimensional Gaussian Random Variable

Let $\delta\mathbf{X}$ denote an $n \times 1$ matrix Gaussian random variable, and let δX denote a particular realization of $\delta\mathbf{X}$. Let us denote the mean value of $\delta\mathbf{X}$ with μ ,

Figure 3.1: Gaussian Density Function $f(x)$

and the covariance matrix of $\delta\mathbf{X}$ with P . Then the Gaussian density function $f(\delta X) \equiv f_{\delta\mathbf{X}}(\delta X)$ on the multidimensional random variable $\delta\mathbf{X}$ is defined by:

$$f(\delta X) = \frac{1}{(2\pi)^{n/2} |P|^{1/2}} \exp\left(-\frac{1}{2}(\delta X - \mu)^T P^{-1}(\delta X - \mu)\right) \quad (3.2)$$

Equation 3.2 is also known as the *multivariate normal* distribution.

3.2 Gaussian Cumulative Distribution Function

The integration of the Gaussian probability density function Eq. 3.1 generates the Gaussian cumulative (probability) distribution function $F(x) \equiv F_{\mathbf{x}}(x)$:

$$F_{\mathbf{x}}(x) = \int_{-\infty}^x f_{\mathbf{x}}(w) dw \quad (3.3)$$

so that:

$$F_{\mathbf{x}}(x) = \Pr\{\mathbf{x} < x\} \quad (3.4)$$

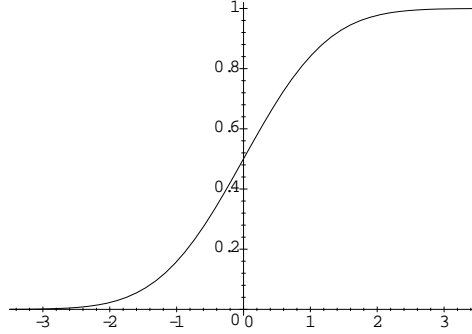
where $\Pr\{\mathbf{x} < x\}$ is the "probability that \mathbf{x} is less than x ."

Figure 3.2 displays $F(x)$. Note that $F(x)$ is *symmetric* about its mean $\mu = 0$:

$$F(x) = 1 - F(-x) \quad (3.5)$$

and that $F(x)$ is *convex* for all $x \leq \mu$ (left-hand convexity)¹, $\mu = 0$:

¹The right-hand side of Ineq. 3.6 specifies an ordinate on a straight line, where that line crosses $F(x)$ at points x_1 and x_2 . Given values for x_1 , x_2 , and λ , then the argument $x = \lambda x_1 + (1 - \lambda)x_2$ for $F(x)$, on the left-hand side of Ineq. 3.6, provides a value for F that

Figure 3.2: Gaussian Distribution Function $F(x)$

$$F(\lambda x_1 + (1 - \lambda) x_2) \leq \lambda F(x_1) + (1 - \lambda) F(x_2) \quad (3.6)$$

where $0 \leq \lambda \leq 1$.

Symmetry and convexity properties are required for any cumulative distribution function that is a candidate for optimal estimation. The Gaussian distribution function satisfies these requirements, but so do many other cumulative distribution functions. Thus, even if real distributions for orbit determination were not quite Gaussian, symmetry and convexity is often satisfied.

3.2.1 Central Limit Theorem

Let random variable \mathbf{x} be the bounded sum of many small independent components with zero means, but where each component may derive from *any* distribution. Then \mathbf{x} will have the Gaussian distribution $F_{\mathbf{x}}(x)$ defined by Eq. 3.3 [22]. The central limit theorem, together with the Fundamental Theorem of Estimation, supports the use of Gaussian distributions for orbit determination.

Application: Thermal Noise

Davenport and Root[99] states (p. 185): "The randomness of the thermally excited motion of free electrons in a resistor gives rise to a fluctuating voltage which appears across the terminals of a resistor. This fluctuation is known as *thermal noise*. Since the total noise voltage is given by the sum of a very large number of electronic voltage pulses, one might expect from the central limit theorem that the total noise voltage would be a Gaussian process. This can indeed be shown to be true."

can be compared to the ordinate on the straight line. Ineq. 3.6 requires that $F(x)$ lie below the straight line when $x \leq \mu$. Symmetry specifies an upside-down convexity for the right-hand side of $F(x)$.

3.2.2 Multidimensional Gaussian Random Variable

The n -fold integration of the Gaussian probability density function Eq. 3.2 generates the Gaussian cumulative distribution function $F(\delta X) \equiv F_{\delta \mathbf{X}}(\delta X)$:

$$F_{\delta \mathbf{X}}(\delta X) = \int_{-\infty}^{\delta X} f_{\delta \mathbf{X}}(w) dw \quad (3.7)$$

so that for each element $F_{\delta \mathbf{X}_j}(\delta X_j)$ of $F_{\delta \mathbf{X}}(\delta X)$:

$$F_{\delta \mathbf{X}_j}(\delta X_j) = \Pr \{ \delta \mathbf{X}_j < \delta X_j \}, \quad j \in \{1, 2, \dots, n\} \quad (3.8)$$

For optimal sequential estimation it is important to note, when $|P| = 0$, that $F_{\delta \mathbf{X}}(\delta X)$ exists even when $f_{\delta \mathbf{X}}(\delta X)$ does not exist. Since P is a symmetric matrix, its determinant is the product of its eigenvalues. When one of its eigenvalues is zero $F_{\delta \mathbf{X}}(\delta X)$ continues to exist, and the optimal sequential filter is stable. But for this case, least-squares estimators fail due to the least-squares requirement for existence of P^{-1} .

3.3 Short Version of Fundamental Theorem

Given that the state estimate error column matrix δX is *Gaussian* and the measurement residual column matrix Δy is *Gaussian*, then the optimal estimator of the state error $\delta \hat{X}$ is the expected value of the state error δX given the measurements of y :

$$\delta \hat{X} = E \{ \delta X | y \} \quad (3.9)$$

That is, the optimal estimator is the conditional mean. For proof, see Meditch [77], Corollary 5.1, page 162.

3.4 Complete Version of Fundamental Theorem

Complete hypotheses for the *Fundamental Theorem of Estimation (Sherman's Theorem)* are satisfied by many cumulative distributions that are not Gaussian [102]. But since there is little justification for us to consider non-Gaussian distributions for the orbit determination problem, the *Short Version* is sufficient. For the *Complete Version* of this important theorem, see Meditch [77], Theorem 5.1, page 160. The original proof of this theorem has been attributed by Kalman to Sherman [101] [102].

3.5 Comparison to Least Squares

For least squares we minimize the sum of squares of weighted measurement residuals. For optimal estimation we minimize the expected value of the root sum of squares on state estimate errors.

Chapter 4

Kalman's Approach

Kalman presented two filtering algorithms, or theorems, for sequential estimation: a *filter measurement-update theorem* and a *filter time-update theorem*.

4.1 Linearization

Kalman's measurement-update theorem requires that the measurement representation be linearly related to the state, and his time update theorem requires linearity in the state transition with time. Unfortunately, orbit determination is nonlinear in the representation of all real measurements from the state, and is nonlinear in the time transition of the complete state. However, the orbit determination problem can be linearized either globally or locally.

Under global linearization, a reference trajectory is propagated from a *single* state estimate and fixed as the global time reference for deriving measurement residuals and state errors. Kalman's linear measurement-update theorem is applied to the mapping from state error to measurement residual, and Kalman's linear time-update theorem is applied to the time transition of state errors. The state error is defined to be the superposition of two differences: the difference between the filter state estimate and the reference-trajectory state estimate, and the difference between the reference-trajectory state estimate and the true (unknown) state. The fixed global reference for a filter may be obtained from a batch least-squares estimate. Global linearization may also be used iteratively by batch least squares.

Local linearization means that *multiple* state estimates are referenced for deriving measurement residuals and state errors.¹ A new state estimate and associated reference trajectory is calculated sequentially at each measurement time. There are as many state estimates and associated reference trajectories as there are measurements. Kalman's linear measurement-update theorem is applied to the mapping from the state error to measurement residual, and

¹Applied to the Kalman filter, local linearization is sometimes called an *extended Kalman filter*.

Kalman's linear time-update theorem is applied to the time transition of state error. The measurement-update theorem and time-update theorem are both applied at the time of the new state estimate. The state error is defined to be the difference between the (unknown) true state and the filter state estimate.

Global linearization erects a formidable barrier not present with local linearization. Namely, the global reference trajectory itself has persistent unknown errors, whereas the errors in the local reference trajectory may be removed (in part) by the next measurement update, and are described by a realistic state error covariance function. The Fundamental Theorem of Estimation is directly applicable to state estimate errors referred to the true unknown state, whereas it is not applicable to state estimate errors referred to a global reference whose errors are unknown and cannot be removed. In the sequel, we shall therefore exclusively use and refer to local linearization.

4.2 State Estimate Error Model

Kalman's state estimate error model is defined by the "formal" linear stochastic differential equation:

$$\frac{d}{dt}\delta X(t) = A(t)\delta X(t) + B(t)\delta\omega(t) \quad (4.1)$$

where $\delta X(t)$ is an $n \times 1$ matrix, where $\delta\omega(t)$ is a $p \times 1$ Gaussian white noise matrix, where $A(t)$ is an $n \times n$ time dependent matrix, and where $B(t)$ is an $n \times p$ time dependent matrix. Comparing Eq. 4.1 to the least squares state estimate error model

$$\frac{d}{dt}\delta X(t) = A(t)\delta X(t)$$

we see that we apparently now have a model to account for acceleration modeling errors; *i.e.*, let $\delta\omega(t)$ denote the resultant of gravity modeling errors, air-drag modeling errors, and solar pressure modeling errors.

Equation 4.1 appears to have great power, but there is a technical difficulty. Namely, $d\delta X(t)/dt$ is undefined because $\delta\omega(t)$ is white noise: the calculus implied by $d\delta X(t)/dt$ is inadequate to describe white noise. Eq. 4.1 can be replaced with a well-defined functional form (see Appendix D), but to carry it through would invoke a rather distracting discussion of *stochastic calculus*, measure theory, and Lebesgue integration (see Bucy and Joseph[10], Chapter 2). Following Kalman [49], we shall refer to Eq. 4.1 as a "formal" linear stochastic differential equation. This means that we shall treat $\delta\omega(t)$ as integrable white noise.

4.3 Integral Equation

When $\delta\omega(t)$ is integrable, one can write the integral to Eq. 4.1 as:

$$\delta X(t_{k+1|k}) = \Phi(t_{k+1}, t_k) \delta X(t_{k|k}) + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) B(\tau) \delta\omega(\tau) d\tau \quad (4.2)$$

Compare Eq. 4.2 to the corresponding least squares integral:

$$\delta X(t_{k+1|k}) = \Phi(t_{k+1}, t_k) \delta X(t_{k|k})$$

to see that Kalman's integral equation is distinguished by the integration of $\delta\omega(\tau)$. One might use this integral to model position and velocity errors due to acceleration errors.

4.3.1 Gauss-Markov Sequence

The state error δX is Gaussian because $\delta\omega$ is Gaussian: a linear transformation on a Gaussian sequence is Gaussian [48]. We need calculate only the first two moments on a Gaussian distribution: its conditional mean (optimal estimate), and its conditional covariance. The entire distribution is thus captured (Theorem 5.3, [77]).

The state error δX is Markov because the transition from $\delta X(t_k)$ to $\delta X(t_{k+1})$ with $t_{k+1} > t_k$ requires no information prior to t_k . This is because $\delta\omega(t)$ is white noise. If $\delta\omega(t)$ were serially correlated, $\delta\omega(t_{k+1})$ would depend on $\delta\omega(t_k)$, $\delta\omega(t_{k-1})$, $\delta\omega(t_{k-2})$, ..., and $\delta X(t_{k+1})$ would depend on $\delta X(t_k)$, $\delta X(t_{k-1})$, $\delta X(t_{k-2})$, ..., requiring one to remember the complete past to move forward. The Gauss-Markov property therefore enables a computationally tractable orbit determination algorithm.

4.4 Time Update Algorithm

4.4.1 State Error Covariance for Filter Time Update

Form the outer product on $\delta X(t_{k+1|k})$ using Eq. 4.2, and take its expectation to define:

$$P_{k+1|k} = E \left\{ \delta X(t_{k+1|k}) \delta X(t_{k+1|k})^T \right\} \quad (4.3)$$

where:

$$\begin{aligned} E \left\{ \delta X(t_{k+1|k}) \delta X(t_{k+1|k})^T \right\} &= \Phi(t_{k+1}, t_k) E \left\{ \delta X(t_{k|k}) \delta X(t_{k|k})^T \right\} \Phi(t_{k+1}, t_k)^T \\ &+ Q_{k+1,k} + I_{k+1,k}^L + I_{k+1,k}^R \end{aligned} \quad (4.4)$$

Since $\delta\omega(t)$ is white, we have:

$$I_{k+1,k}^L = I_{k+1,k}^R = 0 \quad (4.5)$$

$$Q_{k+1,k} = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) B(\tau) E \left\{ \delta\omega(\tau) \delta\omega(\tau)^T \right\} B(\tau)^T \Phi(t_{k+1}, \tau)^T d\tau \quad (4.6)$$

That is:

$$P_{k+1|k} = \Phi(t_{k+1}, t_k) P_{k|k} \Phi(t_{k+1}, t_k)^T + Q_{k+1,k} \quad (4.7)$$

The Kalman filter time-update algorithm is not sufficient for orbit determination. There is a serious problem associated with modeling $\delta\omega(\tau)$ as white noise (Eqs. 4.1, 4.2, and 4.7): gravity modeling errors, air-drag modeling errors, and solar pressure modeling errors are not white. They are serially correlated.

4.5 Measurement Update Algorithm

The Kalman filter measurement-update theorem, and associated algorithm, is derived from Sherman's Theorem. Although its implementation is associated with minor numerical difficulties (*e.g.* [10], p. 141, and Chapter 20), its content has enabled, in part, the realization of optimality. Our presentation employs local linearization, and includes a measurement editing criterion defined by Eqs. 4.11 and 4.15.

Let t_k be the time of the last measurement y_k . Given a new scalar measurement y_{k+1} at time $t_{k+1} \geq t_k$, its scalar (non-zero) measurement error variance R_{k+1} , the state estimate $\hat{X}_{k+1|k}$, and the state estimate error covariance matrix $P_{k+1|k}$, calculate:

$$\Delta y_{k+1} = y_{k+1} - y(\hat{X}_{k+1|k}) \quad (4.8)$$

$$H_{k+1} = \left[\frac{\partial y(X)}{\partial X} \right]_{\hat{X}_{k+1|k}} \quad (4.9)$$

$$\tilde{R}_{k+1} = H_{k+1} P_{k+1|k} H_{k+1}^T + R_{k+1} \quad (4.10)$$

$$\text{If } |\Delta y_{k+1}| < C \sqrt{\tilde{R}_{k+1}}, \text{ Continue} \quad (4.11)$$

$$K_{k+1} = P_{k+1|k} H_{k+1}^T / \tilde{R}_{k+1} \quad (4.12)$$

$$\hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + K_{k+1} \Delta y_{k+1} \quad (4.13)$$

$$P_{k+1|k+1} = (I - K_{k+1} H_{k+1}) P_{k+1|k} \quad (4.14)$$

$$\text{Else discard } y_{k+1}, \text{ acquire } y_{k+2}, k+1 \rightarrow k+2 \quad (4.15)$$

Multiple measurements $y_{k+1} = y_{k+1}^j$, $j \in \{1, 2, \dots\}$, may be processed at time t_{k+1} by repetition of this sequence. Eqs. 4.11 and 4.15 define the Kalman measurement editor, where C is a scalar multiplier prescribing some level of editing (conventionally, $C \approx 3$.)

Notice that \tilde{R}_{k+1} is a scalar: H_{k+1} is $1 \times n$, $P_{k+1|k}$ is $n \times n$ and H_{k+1}^T is $n \times 1$. And note that \tilde{R}_{k+1} is always non-zero because R_{k+1} is always non-zero. Thus the Kalman filter measurement-update algorithm requires *no* calculation of a matrix inverse. Consequently, when any new subset of measurements is unobservable to the state estimate, the filter measurement-update algorithm does not fail.

Chapter 5

Optimal Sequential Filter

An optimal sequential filter is a sequentially processing algorithm that processes forward in time, repeating a recurring, two-step sequence of *time-update* of the state estimate and state covariance, and *measurement-update* of the state estimate and state covariance. The *time update* propagates the state estimate and its error covariance forward, and the *measurement update* incorporates the next measurement. State estimate error magnitudes grow during the time-update interval, and are reduced by the measurement update. The growth of state-estimate error magnitudes is due most significantly to errors in force models: gravity, air-drag, solar photon pressure, spacecraft thrusting and outgassing, and thermal re-radiation. Sequential processing accounts for the recursive growth and reduction of state estimate errors according to the time order in which they are realized, through models based on physical uncertainties.

The measurement update algorithm (Eqs. 4.8 through 4.15), due to Kalman, satisfies Sherman's Theorem. The time update algorithm (Eqs. 5.12 through 5.13), due to Wright, also satisfies Sherman's No matrix inversion is performed in either the filter time-update or filter measurement-update; thus, if the state estimate structure contains unobservable parameters, this condition has no effect on the filter.

5.1 State Estimate Error Model

The model for the state error $\delta X(t)$ is defined by the linear stochastic differential equation:

$$\frac{d}{dt} \delta X(t) = A(t) \delta X(t) + B(t) \delta u(t) \quad (5.1)$$

where $\delta X(t)$ is an $n \times 1$ matrix, where $\delta u(t)$ is a 3×1 matrix-valued, serially correlated Gaussian random error, where $A(t)$ is an $n \times n$ time dependent matrix, and where $B(t)$ is an $n \times 3$ time dependent matrix. For sequential

orbit determination, $\delta u(t)$ is always serially correlated (non-white) due to serial correlation in modeling errors for gravity [119] [120], air-drag, and solar pressure.

5.2 Integral Equation

The solution of Eq. 5.1 has an integral form:

$$\delta X(t_{k+1}) = \Phi(t_{k+1}, t_k) \delta X(t_k) + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) B(\tau) \delta u(\tau) d\tau \quad (5.2)$$

The right-hand side of this equation is the sum of two $n \times 1$ matrices. The first term propagates initial state-estimate errors $\delta X(t_k)$, and the second term accumulates and propagates acceleration modeling errors $\delta u(t)$ (state-error process noise).

5.3 State Error Covariance for Filter Time Update

Form the outer product on $\delta X(t_{k+1|k})$ using Eq. 5.2, and take its expectation to define:

$$P_{k+1|k} = E \left\{ \delta X(t_{k+1|k}) \delta X(t_{k+1|k})^T \right\} \quad (5.3)$$

where:

$$E \left\{ \delta X(t_{k+1|k}) \delta X(t_{k+1|k})^T \right\} = \Phi(t_{k+1}, t_k) E \left\{ \delta X(t_{k|k}) \delta X(t_{k|k})^T \right\} \Phi(t_{k+1}, t_k)^T + I_{k+1,k}^C + I_{k+1,k}^L + I_{k+1,k}^R \quad (5.4)$$

and where:

$$I_{k+1,k}^C = \int \int_{t_k}^{t_{k+1}} H(t_{k+1}, \tau) E \left\{ \delta u(\tau) \delta u^T(t) \right\} H^T(t_{k+1}, t) d\tau dt \quad (5.5)$$

$$I_{k+1,k}^L = \Phi(t_{k+1}, t_k) \int_{t_k}^{t_{k+1}} E \left\{ \delta X(t_{k+1}|t) \delta u^T(t) \right\} H^T(t_{k+1}, t) dt \quad (5.6)$$

$$I_{k+1,k}^R = \int_{t_k}^{t_{k+1}} H(t_{k+1}, \tau) E \left\{ \delta u(\tau) \delta X^T(t_{k+1}|\tau) \right\} d\tau \Phi^T(t_{k+1}, t_k) \quad (5.7)$$

$$H(t, \tau) = \Phi(t, \tau) B(\tau) \quad (5.8)$$

Then:

$$P_{k+1|k} = \Phi(t_{k+1}, t_k) P_{k|k} \Phi(t_{k+1}, t_k)^T + P_{k+1,k}^{\int\int} \quad (5.9)$$

where:

$$P_{k|k} = E \left\{ \delta X(t_{k|k}) \delta X(t_{k|k})^T \right\} \quad (5.10)$$

and:

$$P_{k+1,k}^{\int\int} = I_{k+1,k}^C + I_{k+1,k}^L + I_{k+1,k}^R \quad (5.11)$$

Eq. 5.9 (formally) specifies the method for moving the state error covariance $P_{k|k}$ from time t_k to time t_{k+1} , and for the accumulation of acceleration modeling errors, to produce $P_{k+1|k}$. State error covariance propagation provides the structure $P_{k+1,k}^{\int\int}$ to accommodate random force modeling errors. This term is referred to as *process-noise covariance*. Notice from Eq. 5.9 that the optimal filter covariance time update is time *sequential*. When $P_{k+1,k}^{\int\int}$ is significant, optimal estimation requires a *sequential* time update so as to incorporate $P_{k+1,k}^{\int\int}$ between measurement times t_k and t_{k+1} . The least-squares model

$$P_{k+1|k} = \Phi(t_{k+1}, t_k) P_{k|k} \Phi(t_{k+1}, t_k)^T$$

has no such structure.

Per our definition of optimality, all state-estimate modeling and modeling errors are derived from appropriate force-modeling physics and sensor performance. That is, the contents of $P_{k+1,k}^{\int\int}$ are not arbitrary. We must seek appropriate approximations from the physics of the force-model.

5.4 Approximate Error-Model Solutions

External force-modeling errors $\delta u(t)$ are serially correlated. Formally speaking, this destroys the Markov property that enables computationally tractable orbit determination algorithms. Therefore, $\delta u(t)$ must be identified from appropriate physics and *transformed* from a non-Markov process to a Markov process for each class of external force modeling errors: gravity, air-drag, and solar pressure.

5.4.1 Gravity Solution

An approximate solution to stochastic gravity modeling errors is given by Wright [119].

5.4.2 Air-Drag Solution

An approximate solution to stochastic air-drag modeling errors is presented in Chapter 17.6.

5.4.3 Solar Pressure Solution

An approximate solution to stochastic solar-pressure modeling errors is presented in Chapter 18.

5.5 Filter Time Update Algorithm

Let t_k be the time of the last measurement. Given the state estimate $\hat{X}_{k|k}$, state estimate error covariance matrix $P_{k|k}$, and a new scalar measurement y_{k+1} at time $t_{k+1} \geq t_k$, calculate:

$$\hat{X}_{k+1|k} = \varphi \left\{ t_{k+1}; \hat{X}_{k|k}, t_k, u \left(\hat{X}(t_k), \tau \right), t_{k+1} \leq \tau \leq t_k \right\} \quad (5.12)$$

$$P_{k+1|k} = \Phi_{k+1,k} P_{k|k} \Phi_{k+1,k}^T + P_{k+1,k}^{f,f} \quad (5.13)$$

where $P_{k+1,k}^{f,f}$ is a sum inclusive of gravity acceleration error covariance, air-drag acceleration error covariance, and solar pressure acceleration error covariance.

5.6 Filter Measurement Update

The Kalman filter measurement-update algorithm is defined in Eqs. 4.8 through 4.15.

5.6.1 Measurement Editing

The measurement-update algorithm always processes, or rejects completely, a new measurement; *i.e.*, it does not reconsider a rejected measurement once it has been rejected.

Kalman Measurement Editor Limitation

Given realistic covariance matrices, and the successful completion of filter initialization, the Kalman measurement editor defined by Eqs. 4.11 and 4.15 may still fail given the following situation. When the time-update function propagates the state-error covariance over an extended time interval, the error variances can grow very significantly due to force modeling errors. Large state-error variances are mapped into measurement-error variances according to Eq. 4.10, and this opens widely the editor threshold. If the first observation processed is a significant outlier¹, and if its residual falls within the editor threshold, then

¹An outlier is a "deviate" observation - one so far removed from the majority of data that the validity of its relationship that population must be questioned. We usually identify an outlier as a highly improbable observation relative to the operational state and measurement uncertainties, but often these values have yet to be fully determined during filter initialization.

the outlying observation is accepted and incorporated by the measurement update, and state-error variances are immediately reduced and the state estimate is thereby corrupted. If subsequent observations are otherwise good, it is possible that the measurement editor will reject these good observations. If so, the state-error variances will grow to the point where the filter editor will eventually accept more observations. If these accepted are good observations, then the filter is autonomously reinitialized. A supplementary editor is therefore required to hasten the processing of good measurements soon after the first measurement outlier is mistakenly accepted.

Filter Initialization Problem

Typically the state error covariance is unrealistic during the filter-initialization time interval because the error covariance matrix associated with the *a priori* orbit estimate is unknown. Thus, the measurement editing thresholds are usually too small. In this case, the measurements are thrown out by the Kalman editor, and no measurements can be processed by the filter measurement-update function. A supplementary editor is therefore required to facilitate filter initialization.

5.6.2 Supplementary Editor

Considering Eq. 4.11, let N_{RT} denote the number of observations sequentially rejected in order to trigger an increase in C , and let N_{AT} denote the number of observations sequentially accepted in order to trigger a decrease in C . Integers N_{RT} (default 10) and N_{AT} (default 3) are defined by the analyst. Let $C = C_{\sigma LO}$ (default 3.0) denote the nominal root-variance level for the Kalman editor threshold, and let $C = C_{\sigma HI}$ (default 100.0) denote the root-variance level for the supplementary editor. If N_{RT} measurements are sequentially rejected, then $C = C_{\sigma HI}$ is used if the supplementary editor has been activated by the analyst. If $N_{\sigma HI}$ is in use, and if N_{AT} measurements are sequentially accepted, then C is switched back to $C_{\sigma LO}$. The supplementary editor partly removes the Kalman editor limitation and solves the filter-initialization problem.

5.6.3 Filter Initialization

The supplementary editor should be used for filter initialization, where N_{AT} (default 1000) is set to a large integer value. After filter initialization, the filter restart capability should be used to modify or eliminate the supplementary editor, and the filter should be restarted.

5.6.4 Filter Divergence

Filter divergence is intimately related to both the Kalman measurement editor and the supplementary editor. Let N_D (default 100) denote the number of complete measurement sets sequentially rejected by an editor to define filter

divergence, where the N_D integer value is set by the user. A complete measurement set refers to all measurements defined at the same time tag.

Kalman Measurement Editor

Assuming that the filter has been completely initialized, and given that the supplementary editor is not in use, filter divergence is defined after N_D complete measurement sets have been sequentially rejected.

Supplementary editor

Assuming that the filter has been completely initialized, and given the use of the supplementary editor, filter divergence is defined after $(N_D + N_{RT})$ complete measurement sets have been sequentially rejected.

Chapter 6

Fixed Interval Sequential Smoother

ODTK uses a fixed interval sequential smoother due to Rauch [97][77]. Inputs to the smoother are outputs from the sequential filter. These filter outputs must therefore be stored while running the filter, for use in the smoother. The last filter output is used to initialize the smoother. The last filter output is the first smoother input. The filter runs forward with time. The smoother runs backwards with time.

For this chapter only, we need notation to distinguish state estimates produced by the filter from state estimates produced by the smoother. Both are used in the same equations. Then let \hat{X} and \hat{P} denote state estimate and covariance output by the *filter*, and let \tilde{X} and \tilde{P} denote state estimate and covariance output by the *smoother*. The hats \hat{X} and \hat{P} denote filter, and the tildas \tilde{X} and \tilde{P} denote smoother. Let t_0 denote the first filter time in the fixed interval $\{t_0, t_L\}$, and let t_L , where $t_0 < t_L$, denote the last filter time in the fixed interval $\{t_0, t_L\}$.

6.1 Smoother Initialization

At the last time t_L in the fixed interval $\{t_0, t_L\}$, set:

$$\tilde{X}_{L|L} = \hat{X}_{L|L} \tag{6.1}$$

$$\tilde{P}_{L|L} = \hat{P}_{L|L} \tag{6.2}$$

6.2 Notation for Smoother Nonlinear State Transition

The left-hand side of Eq. 6.3 provides shorthand notation for propagation of the smoothed state estimate $\tilde{X}_{k+1|L}$ backwards in time from t_{k+1} to $t_k < t_{k+1}$ to get $\varphi_k(\tilde{X}_{k+1|L})$:

$$\varphi_k(\tilde{X}_{k+1|L}) = \varphi(t_k; \tilde{X}_{k+1|L}, t_{k+1}, u(\tau); t_k \leq \tau \leq t_{k+1}) \quad (6.3)$$

where $k \in \{L-1, L-2, \dots, 1, 0\}$. Note that:

$$\tilde{X}_{k|L} \neq \varphi_k(\tilde{X}_{k+1|L}) \quad (6.4)$$

In order to accommodate the smoother state estimate transition function for orbit substates, the VOP trajectory propagator will run backwards with time.

6.3 Smoother Sequential Equations

For $k \in \{L-1, L-2, \dots, 1, 0\}$:

6.3.1 Transition Smoothed State Estimate Backwards

$$\varphi_k(\tilde{X}_{k+1|L}) = \varphi(t_k; \tilde{X}_{k+1|L}, t_{k+1}, u(\tau); t_k \leq \tau \leq t_{k+1}) \quad (6.5)$$

6.3.2 Incorporate Filter Estimate and Covariance at Time t_k

$$\tilde{X}_{k|L} = \hat{X}_{k|k} + \hat{P}_{k|k} \left[\hat{P}_{k|k} + \hat{\Phi}_{k+1,k}^{-1} \hat{P}_{k+1,k}^{ff} \left(\hat{\Phi}_{k+1,k}^{-1} \right)^T \right]^{-1} \left[\varphi_k(\tilde{X}_{k+1|L}) - \hat{X}_{k|k} \right] \quad (6.6)$$

6.3.3 Prepare to Calculate Smoother Covariance

$$A_{k,k+1} = \hat{P}_{k|k} \hat{\Phi}_{k+1,k}^T \hat{P}_{k+1|k}^{-1} \quad (6.7)$$

6.3.4 Smoother Covariance

$$\tilde{P}_{k|L} = \hat{P}_{k|k} + A_{k,k+1} \left[\tilde{P}_{k+1|L} - \hat{P}_{k+1|k} \right] A_{k,k+1}^T \quad (6.8)$$

6.3.5 Notes

Inversion of the linear transition matrix is equivalent to reversing the transition time order:

$$\hat{\Phi}_{i,j}^{-1} = \hat{\Phi}_{j,i} \quad (6.9)$$

The matrix subtraction in Eq. 6.8 is delicate; *i.e.*, we must guarantee that $\tilde{P}_{k|L}$ has no negative eigenvalues due to numerical round-off.

6.3.6 Example for $k = L - 1$

$$\varphi_{L-1}(\tilde{X}_{L|L}) = \varphi(t_{L-1}; \tilde{X}_{L|L}, t_L, u(\tau); t_{L-1} \leq \tau \leq t_L)$$

$$\tilde{X}_{L-1|L} = \hat{X}_{L-1|L-1} + \hat{P}_{L-1|L-1} \left[\hat{P}_{L-1|L-1} + \hat{\Phi}_{L,L-1}^{-1} \hat{P}_{L,L-1}^{\hat{f}} \left(\hat{\Phi}_{L,L-1}^{-1} \right)^T \right]^{-1} \left[\varphi_{L-1}(\tilde{X}_{L|L}) - \hat{X}_{L-1|L-1} \right]$$

$$A_{L-1,L} = \hat{P}_{L-1|L-1} \hat{\Phi}_{L,L-1}^T \hat{P}_{L|L-1}^{-1}$$

$$\tilde{P}_{L-1|L} = \hat{P}_{L-1|L-1} + A_{L-1,L} \left[\tilde{P}_{L|L} - \hat{P}_{L|L-1} \right] A_{L-1,L}^T$$

6.4 Filter-Smoother Consistency Test

The filter-smoother consistency test theorem was derived by McReynolds [73] [74] [75]. The filter and smoother state estimate errors are assumed to be multi-dimensional Normal, but each component of the state estimate error is considered independently¹ of other components.

Calculate the $N \times N$ difference matrix $\bar{P}_{k|L}$ between the filtered covariance matrix $\hat{P}_{k|k}$ and the smoothed covariance matrix $\tilde{P}_{k|L}$ for time t_k :

$$\bar{P}_{k|L} = \hat{P}_{k|k} - \tilde{P}_{k|L} \quad (6.10)$$

for each $k \in \{0, 1, 2, \dots, L\}$. The difference matrix $\bar{P}_{k|L}$ should have no negative eigenvalues. Let us denote the square root of the i^{th} main diagonal element of the $N \times N$ difference matrix $\bar{P}_{k|L}$ as $\sigma_{k|L}^i$. Also, let us calculate the $N \times 1$ difference matrix $\bar{X}_{k|L}$ between filtered state estimate $\hat{X}_{k|k}$ and smoothed state estimate $\tilde{X}_{k|L}$ for time t_k :

$$\bar{X}_{k|L} = \hat{X}_{k|k} - \tilde{X}_{k|L} \quad (6.11)$$

¹When all state estimate error components are considered jointly and simultaneously, coupled state estimate error component test acceptance thresholds would be required. Since large error cross-correlation magnitudes are observed, this could provide a useful state component coupled filter-smoother consistency test.

Let us denote the i^{th} element of the $N \times 1$ difference matrix $\bar{X}_{k|L}$ as $\bar{X}_{k|L}^i$. Now let us calculate and graph the ratio :

$$R_{k|L}^i = \bar{X}_{k|L}^i / \sigma_{k|L}^i \quad (6.12)$$

for each $i \in \{1, 2, \dots, N\}$ and for each $k \in \{0, 1, 2, \dots, L\}$.

6.4.1 Test

If for each $i \in \{1, 2, \dots, N\}$ and for each $k \in \{0, 1, 2, \dots, L\}$ we have:

$$\left| R_{k|L}^i \right| \leq 3 \quad (6.13)$$

with probability ~ 0.99 , then McReynolds' filter-smoother test is satisfied for all state components. If for each $i \in \{1, 2, \dots, N\}$ and for each $k \in \{0, 1, 2, \dots, L\}$ we have:

$$\left| R_{k|L}^i \right| > 3 \quad (6.14)$$

then McReynolds' filter-smoother test is failed for all state components. For each i for which inequality 6.13 is satisfied McReynolds' filter-smoother test is satisfied for that state estimate element, and for each i for which inequality 6.14 holds McReynolds' filter-smoother test is failed for that state estimate element.

Region of Validity

The filter-smoother consistency test may fail (*i.e.* $\left| R_{k|L}^i \right| > 3$) past the final measurement. In this case, the smoother covariance is almost equal to the filter covariance and the test becomes undefined because of the small divisor $\sigma_{k|L}^i$ in Eq. 6.12. (Although $\hat{P}_{k|k}$ and $\tilde{P}_{k|L}$ would equal in a formal sense during filter prediction after the last measurement, in practice they can slightly differ due to the accumulation of rounding errors. They can also differ because transitivity of the Gauss-Markov state parameters does not hold when propagating the smoother state and covariance backwards from the end of the prediction interval versus propagating the filter state forwards. It is these small - yet non-zero - differences that result in a large test value.)

6.5 Smoother Matrix Inversions

The smoother state estimate error covariance matrix inversions² of Eqs. 6.6 and 6.7 are required in order to maintain the state estimate size N for the sequential filter.

²It is possible to avoid matrix inversion in the smoother at the cost of increasing state size by N for each time increment in the smoothing lag. Example: A lag of one time increment requires state size by $2N$.

Chapter 7

Variable-Lag Sequential Smoother

7.1 Introduction

We have developed solutions to the estimation problems associated with near-real-time orbit determination that are defined in sections *General Application* and *Impulsive Maneuvers*. See the section *Required Properties* for the estimator we have developed. See the section *Properties Available* for choices we had. Our new estimator combines a real-time extended Kalman filter (EKF) with a fixed epoch smoother¹ (FES), where the fixed epoch lags EKF measurement time-tags with variable time lag. Thus the name *variable lag smoother* (VLS).

We have implemented two forms of the FES, the Frazer form (FES/F), and the Carlton-Rauch form (FES/CR). FES/F is free of state-sized matrix inverse calculation, but FES/CR requires the calculation of a state-sized covariance matrix inverse for each FES/CR execution.

Orbit Determination Tool Kit (ODTK) previously and currently provides a forward-running EKF, followed by a backward-running fixed-interval smoother (FIS). Two separate user operations are required to run the EKF-FIS filter-smoother. Fortunately the EKF runs in real-time and does *not* require calculation of a state-sized covariance matrix inverse. Unfortunately the FIS *does* require calculation of a state-sized covariance matrix inverse, and is not applicable to near-real-time operations.

Our new forward running VLS does *not* require calculation of a state-sized covariance matrix inverse for FES/F, provides results in near-real-time, and combines filter and smoothing calculations with one user selected operation.

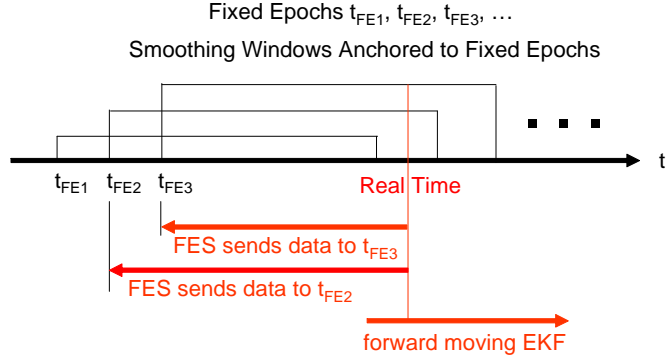


Figure 7.1: Variable Lag Smoother with EKF and FES

7.2 General Application

Figure 7.1 illustrates a general application of our VLS with fixed epochs on a uniform time grid. A smoother window is anchored to each fixed epoch. Smoother window definition logic is selected by the user at run-time. The most simple user option is to select fixed constant window length for each smoothing window. Or the user may select time-varying criteria that require satisfaction of accuracy thresholds by time-varying covariance matrix elements, together with maxima of smoother window length. Smoother windows may be overlapping as in Figure 7.1, or they may be non-overlapping, and possibly on a non-uniform time grid, as in Figure 7.2.

The EKF moves on a time grid dictated by measurement time-tags, generally non-uniform. When the EKF reaches a fixed epoch, a new FES is created. The FES estimate is initialized by the EKF estimate at the fixed epoch after all measurements have been processed by the EKF at the fixed epoch. As the EKF epoch moves forward, information derived from the EKF at each measurement time-tag is sent backwards to the fixed epoch by the FES while the EKF epoch is in the smoother window. The FES is invoked only for EKF measurement updates. The FES is not invoked for EKF time updates, unlike FIS operation. When the EKF epoch exits the smoother window the FES is destroyed, and associated smoother results are recorded. The lag of fixed epoch relative to the EKF epoch increases throughout passage of the smoother window by the EKF due to forward motion of the EKF epoch.

When a VLS computer run would be terminated with the EKF epoch inside of a smoothing window, the user will have an option to accept results of the VLS with partial completion of the last smoothing window, or to save results into an EKF restart file just prior to entry of the EKF epoch into the last smoothing window.

¹Our *fixed epoch* is the *fixed point* referred to for the fixed point smoother[77].

7.2.1 ODTK LEO Simulation

Results

ODTK intrack position results, from processing simulated two-way range measurements from eight AFSCN ground stations and a LEO spacecraft with transponder, were presented in July 2008[131]. We showed consistency of the VLS covariance matrix function with estimation errors, and accuracy equivalence of the VLS with the FIS. We presented contrasting accuracy performance improvement of the VLS FES relative to the VLS EKF. We showed that the new VLS enables trading maximum lag time for accuracy performance.

7.3 Impulsive Maneuvers

Consider a single spacecraft that performs impulsive² maneuvers, and consider the sequential estimation of the state from tracking measurements. The state includes the six components of position and velocity, force model parameters, and time-varying measurement biases.

ODTK does successfully estimate velocity change due to an impulsive maneuver using the FIS, but this is an off-line operation, relative to the ODTK EKF. The FIS estimation runs backward with time.

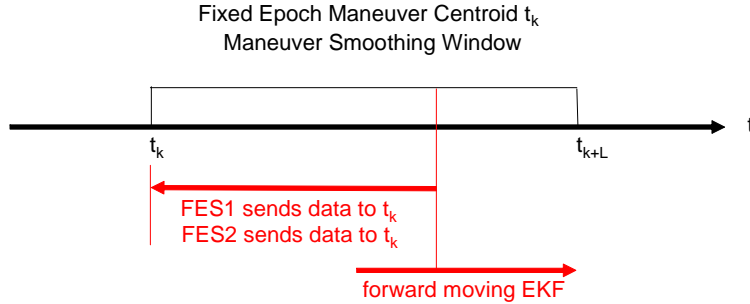


Figure 7.2: Estimation of Velocity at Fixed Epochs with EKF and FES

Our new VLS method eliminates the off-line FIS, uses a forward running EKF, and uses two FES smoothers that send EKF estimate information backwards to a fixed epoch. The fixed epoch is defined by an impulsive maneuver time centroid, say t_k as illustrated in Figure 7.2.

Let t_k denote the time centroid of an impulsive spacecraft maneuver. Let $\hat{X}_{k|i}^{(-)}$ and $P_{k|i}^{(-)}$ denote the state estimate and covariance derived by the EKF from processing the last measurement y_i with time-tag $t_i \leq t_k$. The estimate

²Impulsive maneuvers are used to model thrust intervals that are short as compared to orbit period.

$\hat{X}_{k|i}^{(-)}$ does *not* include addition of the velocity change at t_k , and its associated covariance matrix $P_{k|i}^{(-)}$ does not include the addition of EKF maneuver process noise covariance at t_k . The EKF state estimate $\hat{X}_{k|i}^{(-)}$ and covariance matrix $P_{k|i}^{(-)}$ are used to initialize FES1 at time t_k .

Let $\hat{X}_{k|i}^{(+)}$ and $P_{k|i}^{(+)}$ denote the state estimate and covariance derived after application of velocity change to $\hat{X}_{k|i}^{(-)}$, and after addition of maneuver process noise covariance to $P_{k|i}^{(-)}$. The state estimate $\hat{X}_{k|i}^{(+)}$ and covariance matrix $P_{k|i}^{(+)}$ are used to initialize FES2 at time t_k . A smoothing window $\{t_k, t_{k+L}\}$ is selected by the user, explicitly or implicitly. Now we process measurements y_j , $j = k + 1, k + 2, \dots, k + L$ by the EKF, and operate both FES1 and FES2 on the EKF output across the smoothing window $\{t_k, t_{k+L}\}$. The VLS with two FES smoothers produces smoothed estimates $\hat{X}_{k|k+L}^{(-)}$ and $\hat{X}_{k|k+L}^{(+)}$, and their covariance matrices $P_{k|k+L}^{(-)}$ and $P_{k|k+L}^{(+)}$. The desired estimate of velocity change is found in the velocity components of the difference $\hat{X}_{k|k+L}^{(+)} - \hat{X}_{k|k+L}^{(-)}$.

7.4 Required Properties

Let A denote the class of *linear* optimal sequential estimators presented by Meditch[77] with members

- Kalman filter (KF)
- fixed-interval smoother
- Carlton-Rauch fixed-epoch smoother
- Frazer fixed-epoch smoother
- fixed-lag smoother
- combination of above

Let \tilde{A} denote the class of *nonlinear* optimal sequential estimators that derive from the linear estimators of class A by extension. And let \check{A} denote the union of classes A and \tilde{A} . For prototype in Class \check{A} we have the well-known extended Kalman filter (EKF). Epoch for the KF and EKF is defined by the time-tag t_k of last measurement processed, or by propagation to $t_{k+1} > t_k$. But the epoch for each smoothed state estimate of interest lags the time-tag t_k of last measurement processed by the KF. The state estimate error magnitude is always smallest, thus accuracy is best, at estimation epoch³ for each member of Class \check{A} . But each smoothed state estimate has better accuracy than a filtered state

³In contrast, note that state estimate error magnitude for each iterative batch least squares (LS) estimate depends on LS data fit span and distribution of data in the fit span. Thus LS state estimate error magnitude is independent of epoch placement.

estimate, at the same epoch, when the smoothed state estimate epoch lags KF measurement time-tags with a significant time-lag. Here the smoother has processed more measurements than the filter, and the smoother estimate derives from information forward of the smoother epoch, and from information prior to the smoother epoch, whereas the filter estimate derives only from information prior to the same epoch. The latter has special significance.

There exists a class of nonlinear estimation problems that cannot be solved with a real-time extended Kalman filter (EKF) because all tracking data precedes the EKF state epoch, and tracking data following the real-time state epoch are not available. An example is provided by the problem of estimating the velocity change due to an impulsive maneuver at the time of maneuver impulse. The most accurate state estimate must use tracking data both before and after the time of maneuver impulse. A second example is provided by the optimal simultaneous estimation of orbit and atmospheric density with a sequential estimator where use is made of tracking data before the state epoch and tracking data after the state epoch. A third example is provided by the optimal GPS orbit determination and time-transfer problem using GPS pseudo-range measurements and GPS carrier-phase measurements, where carrier-phase range-biases must be estimated at fixed epochs. These examples demonstrate the need for a smoothing lag that is time-variable. Sequential smoothers are useful in solving this class of problems.

ODTK uses an EKF that runs forward with time followed by an extended FIS that runs backward with time. But this filter-smoother of Class \tilde{A} cannot be conveniently used for *near-real-time* estimation because all measurements of interest across the fixed-interval must be completely filtered and smoothed before a smoothed state estimate is available for use. We appear to need a smoother from Class \tilde{A} that runs forward with time to achieve near-real-time orbit determination. Thus we desire an estimator with properties

1. Class \tilde{A}
2. runs forward with time (pause acceptable)
3. near-real-time throughput
4. accuracy performance due to optimal smoothing
5. smoothing lag is time-variable

7.5 Properties Available

7.5.1 Linear

We have studied smoothers of Class A for extension to Class \tilde{A} . Choices are best understood by reviewing particular aspects of Class A estimators for discrete linear systems. The best presentation of Class A estimators was given by Meditch[77], unified theoretically by use of Sherman's Theorem[101][102][77][125],

and unified notationally with adoption of Kalman's indexing[49][77]. We choose *discrete* linear systems in preference to *continuous* linear systems because the trajectory measurements for orbit determination of space objects are always discrete. We refer to these linear algorithms with acronyms identified by Table 7.1. These algorithms have properties summarized in Table 7.2, where FE refers to the fixed epoch of a fixed epoch smoother.

Class A	Acronym
Kalman Filter	KF
Fixed Interval Smoother	FIS
Fixed Epoch Smoother/Carlton-Rauch	FES/CR
Fixed Epoch Smoother/Frazer	FES/F
Fixed Lag Smoother	FLS

Table 7.1: Acronyms for Linear Sequential Estimators

7.5.2 Extension

State estimates of Class A are always propagated with a linear transition matrix function Φ . It is important to note that extension from Class A to Class \tilde{A} can be achieved with two very different techniques. First, propagate *variations* of the state estimate with a linear transition matrix function Φ . It appears that this can always be achieved. Or second, propagate the state estimate directly with a nonlinear transition function φ . It appears that this cannot always be achieved with sequential smoothers.

Class A	KF	FIS	FES/CR	FES/F	FLS
Initial Conditions	user	terminal KF	KF at FE	KF at FE	after KF-FIS
Time Direction	forward	backward	forward	forward	forward
Throughput	real-time	after KF-FIS	near real-time	near real-time	near real-time
Matrix Inverses	0	1	1	0	3
Extension φ	yes	yes	no	no	no
Extension Φ	yes	yes	yes	yes	yes

Table 7.2: Available Properties of Linear Estimators

7.5.3 Nonlinear

Extension φ of Table 7.2 refers to an acceptable conversion of every linear propagation of state estimate to nonlinear numerical integration, for extension from Class A to Class \tilde{A} . We have successfully used numerical integration φ for

extension of KF and FIS algorithms for many years. But this does not appear to be possible for FES/CR, FES/F and FLS algorithms.

Extension Φ of Table 7.2 refers to the acceptable use of a linear transition matrix function to propagate state estimate *variations*, rather than state estimates, for extension from Class *A* to Class \hat{A} . The FES/CR algorithm is our first successful example. The EKF measurement residual is a linear variation used successfully without propagation in the EKF. The EKF measurement residual has been successfully propagated linearly in the FES/F algorithm.

The FLS algorithm is rejected for use in our VLS because it does not satisfy Property 5: *smoothing lag is time-variable*. Also, the FLS algorithm is undesirable because it requires the calculation of three state-sized matrix inverses. The FES/F is attractive for use in our VLS because no state-sized matrix inverse is required.

7.6 Kalman Filter

7.6.1 Time Update

Linear

The Kalman filter linear algorithm is derived and presented by Meditch[77] in Theorem 5.5 page 176. Let t_k be the time of last measurement. We are given the

- $n \times 1$ matrix state estimate $\hat{X}_{k|k}$
- $n \times n$ matrix state estimate error covariance matrix $P_{k|k}$
- $n \times p$ disturbance transition matrix $\Gamma_{k+1,k}$
- $p \times p$ process noise covariance matrix Q_k
- new measurement y_{k+1} at time $t_{k+1} > t_k$

Calculate the propagated state estimate $\hat{X}_{k+1|k}$ and covariance $P_{k+1|k}$

$$\hat{X}_{k+1|k} = \Phi_{k+1,k} \hat{X}_{k|k} \quad (7.1)$$

$$P_{k+1|k} = \Phi_{k+1,k} P_{k|k} \Phi_{k+1,k}^T + \Gamma_{k+1,k} Q_k \Gamma_{k+1,k}^T \quad (7.2)$$

Nonlinear

Here we present our particular form for an extended Kalman filter (EKF) Time Update. State estimate propagation is nonlinear, so the linear propagation $\Phi_{k+1,k} \hat{X}_{k|k}$ of Equation 7.1 must be replaced with a numerical integrator $\varphi \{ \cdot \}$

$$\hat{X}_{k+1|k} = \varphi \left\{ t_{k+1}; \hat{X}_{k|k}, t_k, u \left(\hat{X}(\tau|t_k), \tau \right), t_{k+1} \leq \tau \leq t_k \right\} \quad (7.3)$$

and the propagation $\Gamma_{k+1,k} Q_k \Gamma_{k+1,k}^T$ of white noise covariance Q_k must be replaced with a physically connected non-white noise covariance $P_{k+1,k}^{\int \int}$

$$P_{k+1|k} = \Phi_{k+1,k} P_{k|k} \Phi_{k+1,k}^T + P_{k+1,k}^{\int \int} \quad (7.4)$$

where $P_{k+1,k}^{\int \int}$ is composed of a sum of doubly integrated acceleration error covariance functions due to gravity, air-drag, solar pressure, and thrusting.

7.6.2 Measurement Update

Linear

Let t_k be the time of last measurement y_k . Given a new scalar measurement y_{k+1} at time $t_{k+1} > t_k$, its non-zero measurement error covariance R_{k+1} , the propagated state estimate $\hat{X}_{k+1|k}$, and the propagated state estimate error covariance matrix $P_{k+1|k}$, and the measurement-state $1 \times n$ row matrix H_{k+1} , calculate

$$\Delta y_{k+1|k} = y_{k+1} - H_{k+1} \hat{X}_{k+1|k} \quad (7.5)$$

$$\tilde{R}_{k+1} = H_{k+1} P_{k+1|k} H_{k+1}^T + R_{k+1} \quad (7.6)$$

$$K_{k+1} = P_{k+1|k} H_{k+1}^T \tilde{R}_{k+1}^{-1} \quad (7.7)$$

$$\hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + K_{k+1} \Delta y_{k+1|k} \quad (7.8)$$

$$P_{k+1|k+1} = (I - K_{k+1} H_{k+1}) P_{k+1|k} \quad (7.9)$$

Nonlinear

Class \tilde{A} is distinguished with the nonlinear measurement representation $y(\hat{X}_{k+1|k})$ and calculations

$$H_{k+1} = \left[\frac{\partial y(X)}{\partial X} \right]_{\hat{X}_{k+1|k}} \quad (7.10)$$

$$\Delta y_{k+1|k} = y_{k+1} - y(\hat{X}_{k+1|k}) \quad (7.11)$$

But calculations for \tilde{R}_{k+1} , K_{k+1} , $\hat{X}_{k+1|k+1}$, and $P_{k+1|k+1}$ have the same form as for Class A

$$\tilde{R}_{k+1} = H_{k+1} P_{k+1|k} H_{k+1}^T + R_{k+1} \quad (7.12)$$

$$K_{k+1} = P_{k+1|k} H_{k+1}^T \tilde{R}_{k+1}^{-1} \quad (7.13)$$

$$\hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + K_{k+1}\Delta y_{k+1|k} \quad (7.14)$$

$$P_{k+1|k+1} = (I - K_{k+1}H_{k+1})P_{k+1|k} \quad (7.15)$$

Our ODTK EKF design has been successfully applied to many orbit determination problems. References [122], [123], [126], [127], and [128] exemplify what we mean by physically connected state estimate error covariance function. When processing GPS pseudo-range and carrier phase measurements, an intimate understanding and use of the stochastic GPS composite clock is required, as demonstrated in References [130] and [129]. In summary, our definition of EKF is uniquely specialized by the stochastic physics encountered in the orbit determination problem.

7.7 Carlton-Rauch Fixed-Epoch Smoother

7.7.1 FES Initialization

Let t_{FE} denote a fixed epoch, coincident with time centroid of an impulsive spacecraft maneuver and known a priori. Let \hat{X}_{FE} and P_{FE} denote filtered state estimate and covariance at t_{FE} . Meditch's presentation of the Carlton-Rauch FES denotes t_{FE} as coincident with time-tag of some measurement processed; *i.e.*, $t_{\text{FE}} = t_k$ for $\hat{X}_{\text{FE}} = \hat{X}_{k|k}$ and $P_{\text{FE}} = P_{k|k}$, but this is not necessary⁴. It may be necessary that $\hat{X}_{\text{FE}} = \hat{X}_{k|k-1}$ and $P_{\text{FE}} = P_{k|k-1}$ for propagated state estimate $\hat{X}_{k|k-1}$ and covariance $P_{k|k-1}$. However, to be consistent with Meditch's presentation I shall continue with his notation.

Let $\hat{X}_{k|j-1}$ denote an FES state estimate with fixed epoch t_k where $j = k+1$, where the last measurement processed by the filter has time-tag t_k or t_{k-1} , and where $t_k \leq t_{j-1}$ or $t_{k-1} \leq t_{j-1}$ respectively. With the filter at time $t_{\text{FE}} = t_k$, initialize the FES by storing objects associated with, or calculated by, the filter. Store t_k and

$$\hat{X}_{k|j-1} = \hat{X}_{\text{FE}} \quad (7.16)$$

$$P_{k|j-1} = P_{\text{FE}} \quad (7.17)$$

$$\hat{X}_{j-1|j-1} = \hat{X}_{\text{FE}} \quad (7.18)$$

$$P_{j-1|j-1} = P_{\text{FE}} \quad (7.19)$$

$$B_{j-1} = I \quad (7.20)$$

⁴If one processes a pseudo-measurement with time-tag t_{FE} that has zero measurement-state partial derivatives, then Meditch's notation is maintained and no harm is done to the VLS.

7.7.2 Measurement at $t_j = t_{k+1}$

For this section $j = k + 1$, where t_k is the fixed epoch and $t_j > t_k$ is the time-tag for a new measurement $y_j = y_{k+1}$.

Filter

The filter calculates the propagated state estimate $\hat{X}_{k+1|k} = \hat{X}_{j|j-1}$, propagated covariance $P_{k+1|k} = P_{j|j-1}$, filtered state estimate $\hat{X}_{k+1|k+1} = \hat{X}_{j|j}$, filtered covariance $P_{k+1|k+1} = P_{j|j}$, and transition matrix $\Phi_{k+1,k} = \Phi_{j,j-1}$. Store $\hat{X}_{j|j-1}$, $P_{j|j-1}$, $\hat{X}_{j|j}$, $P_{j|j}$, and $\Phi_{j,j-1}$ for use in the FES. For the first value of B_j , following FES initialization, set

$$B_j = B_{j-1}A_{j-1} \quad (7.21)$$

where

$$A_{j-1} = P_{j-1|j-1}\Phi_{j,j-1}^T P_{j|j-1}^{-1} \quad (7.22)$$

FES

FES calculations refer to the fixed epoch t_k , and to filter measurement time-tags $t_j \geq t_k$.

$$\hat{X}_{k|j} = \hat{X}_{k|j-1} + B_j \left[\hat{X}_{j|j} - \hat{X}_{j|j-1} \right] \quad (7.23)$$

$$P_{k|j} = P_{k|j-1} + B_j \left[P_{j|j} - P_{j|j-1} \right] B_j^T \quad (7.24)$$

If the column matrix $\hat{X}_{k|j}$ has n elements, then $P_{k|j}$, $P_{k|j-1}$, $P_{j|j-1}$, $P_{j|j}$, B_j , and $\Phi_{j,j-1}^T$ are $n \times n$ matrices. Covariance matrices $P_{k|j}$, $P_{k|j-1}$, $P_{j|j-1}$, and $P_{j|j}$ are symmetric and are free of negative eigenvalues. Zero eigenvalues in $P_{j|j-1}$ are not acceptable because it must be inverted. The implementation must guarantee that symmetric matrices are numerically symmetric, that all covariance matrices are numerically free of negative eigenvalues, and that $P_{j|j-1}$ is free of zero eigenvalues.

Filter

After FES execution and recording of FES results, the FES recursion is performed by the filter in preparation for the next measurement.

$$\hat{X}_{k|j-1} = \hat{X}_{k|j} \quad (7.25)$$

$$P_{k|j-1} = P_{k|j} \quad (7.26)$$

$$B_{j-1} = B_j \quad (7.27)$$

7.7.3 Measurements at $t_j = t_{k+1}, t_{k+2}, \dots$

In the section *Measurement at $t_j = t_{k+1}$* above, replace t_{k+1} with t_{k+2} for the measurement $y_j = y_{k+2}$ at time $t_j = t_{k+2}$. When $t_j = t_{k+h}$, replace t_{k+1} with t_{k+h} for the measurement $y_j = y_{k+h}$ at time $t_j = t_{k+h}$.

7.8 Frazer Fixed-Epoch Smoother

7.8.1 FES Initialization

Let t_{FE} denote a fixed epoch, coincident with time centroid of an impulsive spacecraft maneuver and known a priori. Let \hat{X}_{FE} and \hat{P}_{FE} denote filtered state estimate and covariance at t_{FE} . Meditch's presentation ([77] Corollary 6.1 page 232) of the Frazer FES denotes t_{FE} as coincident with time-tag of some measurement processed; *i.e.*, $t_{\text{FE}} = t_k$ for $\hat{X}_{\text{FE}} = \hat{X}_{k|k}$ and $P_{\text{FE}} = P_{k|k}$, but this is not necessary⁵. It may be necessary that $\hat{X}_{\text{FE}} = \hat{X}_{k|k-1}$ and $P_{\text{FE}} = P_{k|k-1}$ for propagated state estimate $\hat{X}_{k|k-1}$ and covariance $P_{k|k-1}$. However, to be consistent with Meditch's presentation I shall continue with his notation.

Let $\hat{X}_{k|j-1}$ denote an FES state estimate with fixed epoch t_k where $j = k+1$, where the last measurement processed by the filter has time-tag t_k or t_{k-1} , and where $t_k \leq t_{j-1}$ or $t_{k-1} \leq t_{j-1}$ respectively. With the filter at time $t_{\text{FE}} = t_k$, initialize the FES by storing objects associated with, or calculated by, the filter: t_k , $\hat{X}_{k|j-1} = \hat{X}_{\text{FE}}$, $P_{k|j-1} = P_{\text{FE}}$, and $W_{j-1} = P_{\text{FE}}$.

7.8.2 Measurement at $t_j = t_{k+1}$

For this section $j = k+1$, where t_k is the fixed epoch and $t_j > t_k$ is the time-tag for a new measurement $y_j = y_{k+1}$.

Filter

The filter calculates the propagated state estimate $\hat{X}_{k+1|k} = \hat{X}_{j|j-1}$, propagated covariance $P_{k+1|k} = P_{j|j-1}$, filtered state estimate $\hat{X}_{k+1|k+1} = \hat{X}_{j|j}$, filtered covariance $P_{k+1|k+1} = P_{j|j}$, transition matrix $\Phi_{k+1,k} = \Phi_{j,j-1}$, measurement-state Jacobian matrix $H_{k+1} = H_j$, measurement covariance matrix $R_{k+1} = R_j$, and measurement residual $\Delta y_{k+1,k} = \Delta y_{j,j-1}$ at time-tag $t_{k+1} = t_j$ for the new measurement $y_{k+1} = y_j$. Store $\hat{X}_{j|j-1}$, $P_{j|j-1}$, $X_{j|j}$, $P_{j|j}$, $\Phi_{j,j-1}$, H_j , R_j , and $\Delta y_{j,j-1}$ for use in the FES.

FES

The following algorithm was constructed by Frazer ([77] Corollary 6.1 page 232). Fixed epoch smoother calculations refer to the fixed epoch t_k , and to

⁵If one processes a pseudo-measurement with time-tag t_{FE} that has zero measurement-state partial derivatives, then Meditch's notation is maintained and no harm is done to the VLS.

filter measurement time-tags $t_j \geq t_k$.

$$S_j = H_j^T R_j^{-1} H_j \quad (7.28)$$

$$W_j = W_{j-1} \Phi_{j,j-1}^T [I - S_j P_{j|j}] \quad (7.29)$$

$$\hat{X}_{k|j} = \hat{X}_{k|j-1} + W_j H_j^T R_j^{-1} \Delta y_{j,j-1} \quad (7.30)$$

$$P_{k|j} = P_{k|j-1} - W_j [S_j P_{j|j-1} S_j + S_j] W_j^T \quad (7.31)$$

If the column matrix $\hat{X}_{k|j}$ has n elements, then $P_{k|j}$, $P_{k|j-1}$, $P_{j|j-1}$, $P_{j|j}$, $\Phi_{j,j-1}^T$, S_j , and W_j are $n \times n$ matrices. Covariance matrices $P_{k|j}$, $P_{k|j-1}$, $P_{j|j-1}$, and $P_{j|j}$ are symmetric matrices with positive or zero eigenvalues. Zero covariance matrix eigenvalues are acceptable because no state-sized covariance matrix inverse is required for Frazer form of the FES. S_j is seen to be symmetric by inspection of its defining Equation 7.28. The implementation must guarantee that symmetric matrices are numerically symmetric, and that covariance matrices are numerically free of negative eigenvalues. W_{j-1} is initialized as a symmetric covariance matrix, but W_j and subsequent W_{j-1} matrices are not symmetric due to the factor $\Phi_{j,j-1}^T$ in the recursive Equation 7.29.

FES calculations require products of matrices with extreme differences in order of magnitude. For example, the calculation of W_j according to Equation 7.29 requires evaluation of the product $S_j P_{j|j}$ that is subtracted from a matrix of order unity. The eigenvalues of S_j are very large due to the small values of R_j , and the eigenvalues of $P_{j|j}$ are very small, all non-negative. The product $S_j P_{j|j}$ is of order unity, but some significance is lost in double precision calculations. It may thus be advisable to premultiply S_j by a small positive scalar ϵ and premultiply $P_{j|j}$ by its inverse ϵ^{-1} for calculation of the product $S_j P_{j|j} = (S_j \epsilon) (\epsilon^{-1} P_{j|j})$, where $(S_j \epsilon)$ and $(\epsilon^{-1} P_{j|j})$ are both of order unity.

Filter

After FES execution and recording of FES results, the FES recursion is performed by the filter in preparation for the next measurement.

$$\hat{X}_{k|j-1} = \hat{X}_{k|j} \quad (7.32)$$

$$P_{k|j-1} = P_{k|j} \quad (7.33)$$

$$W_{j-1} = W_j \quad (7.34)$$

Chapter 8

State-Space Changes

8.1 Problem Definition

Let us consider a real-time Variable Lag Smoother(VLS) [131] that produces simultaneously both filtered and smoothed estimates and covariances. The VLS provides smoothed estimates on a set of fixed epochs in the past (relative to the current filter epoch) within a moving window, not to exceed a specified lag (the maximum lag). Heretofore, the VLS algorithms assume that the dimension of the state of the system is fixed and there is no state-space discontinuity. It is now shown how VLS algorithms can handle these state space discontinuities, including adding and subtracting states and handling discontinuities in states.

The epoch of a filter will increase with time along with the time tags associated with the data. The filter epochs may also include other times that do not necessarily coincide with data tags. At each point, the filter will produce an estimate and covariance ordered in increasing time. These quantities, along with possibly other calculated quantities, are stored for the standard fixed-interval smoothers that process this data backwards. The smoother will usually operate on the same time grid as the filter. (An exception to this is the fixed point smoother, where the smoother only corrects the state estimate at a single epoch.) The VLS will only correct the state estimate on a specified grid, which is a subset of the filter grid.

A process that may be imposed upon real-time filters and smoothers is a monitoring. In monitoring, the user wants to know what the solution (state estimate and covariance) looks like on a regular real-time grid that is chosen to be different than the filter grid. For example, points on the filter grid may differ by 1 minute. However, the user wants to examine the changes of the smooth points every 2 minutes. The monitoring grid may add points to the grids of both filters and smoothers, if the monitoring grid does not coincide with the filter grid. Monitoring may also force the smoothing algorithms to repeatedly generate smoothed solutions for the same monitoring epochs more than once, driving up the CPU requirements. The number of smoothing operations is

discussed below as a function of matrix dimensions and grid points per second.

8.1.1 Structure of VLS Algorithms

Based upon the above problem description, the algorithm for solution has the following structure:

1. A filter processes data and VLS algorithm updates smoothed estimates and covariances on an output grid. As the filter progresses, a point will be added to smoother grid when it enters the smoothing window, and smoother grid point will be dropped when it leaves the smoother window.
3. When an epoch of the output monitoring grid is reached, all points on the output grid whose estimates are being updated will have up-to-date estimates and covariances produced using all the data up to the epoch time of the filter.

Various smoothing algorithms, as mentioned above, can be used to address this problem. We wish to address the case where the dimension of the filter varies, i.e. the dynamic state space can *expand* or *contract*. An epoch at which the state size changes shall be referred to as a *filter discontinuity*. Discontinuity can also occur if the dimension does not change but the states are *discontinuous*, such as when the new states represent a different object than the old state. This latter case is equivalent to first contracting, then expanding, the state at a specific epoch.

8.2 Handling Discontinuities by the Filter

Discontinuities are naturally introduced in many of applications, such as when more than one object is being tracked. For example, tracking data for a spacecraft may be suspended, such that the filter would do well to drop this spacecraft's state from the filter. Conversely, one may desire to add the state of a newly tracked spacecraft to the filter. Other cases of discontinuities can occur within the state of a single spacecraft, e.g. an orbit maneuver or a clock reset or change, an applied acceleration, etc. In this case new states must replace the old ones in the filter, such that the filter must first contract and then expand, leaving the dimension of the filter state unchanged.

8.2.1 State Contraction

Deletions and contractions are performed from the state vector estimate and the covariance, i.e. deleting rows and columns of the covariance of the deleted state variables. Expressed mathematically, let \hat{X} be the dynamic state vector estimate, which can be decomposed into two components:

$$\hat{X} = \begin{bmatrix} \hat{Y} \\ \hat{Z} \end{bmatrix} \quad (8.1)$$

The state-error covariance of \hat{X} is partitioned as follows:

$$P_{XX} = \begin{bmatrix} P_{yy} & P_{yz} \\ P_{zy} & P_{zz} \end{bmatrix} \quad (8.2)$$

Here, $P_{zy} = P_{yz}^T$. For state contraction, let us delete \hat{Z} from the state, so that the filter state vector and covariance becomes:

$$\hat{X} = \hat{Y} \quad (8.3)$$

$$P_{XX} = P_{yy} \quad (8.4)$$

In practice, the deleted states may not be at the end of the state vector. When this is the case, it is necessary to modify the state vector by deleting the eliminated elements and collapsing the state vector to having only those variables that are being estimated. Similar operations must be done to the covariance: rows and columns of the covariance corresponding to the deleted states are also deleted to maintain a square matrix whose collapsed dimension is compatible with the contracted state vector. Mathematically, this can be expressed by the operator Q , an example of which given below.

Let us delete the second variable of a 5-state vector and reverse the order of the 3rd and 4th vector. A Q operator that does this is given by

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (8.5)$$

Multiply the state by this Q operator to the column state vector X to generated a four-dimensional state vector X' :

$$X' = Q X \quad (8.6)$$

Define the $n \times n$ reorder operator (n being the dimension of the state) such that:

$$Q(i, j) = 1 \text{ if the } j\text{th parameter becomes the } i\text{th (0 otherwise)} \quad (8.7)$$

In general, the $p \times n$ operator Q will delete $n-p$ states and reorder the remaining states. For every variable that is to be retained, a 1 in the row corresponds to the new location, with the column corresponding to its original position; all other terms in this row are 0. This operator not only selects the states but can also generate a specific ordering of the selected states.

The covariance also needs to be collapsed and reordered. Here the same operator is applied to both the rows and columns; mathematically, this can be expressed by the equation:

$$P' = Q P Q^T \quad (8.8)$$

Applying the operator Q to the left selects and orders the rows of the covariance; applying Q to the right of P selects the same column and gives the same order

so that the final matrix P' is the covariance of the errors in X' . Using the above definition, the 5×5 covariance P is compressed to the 4×4 covariance P' :

$$P' = \begin{bmatrix} P_{11} & P_{14} & P_{13} & P_{15} \\ P_{41} & P_{44} & P_{43} & P_{35} \\ P_{41} & P_{34} & P_{44} & P_{45} \\ P_{51} & P_{54} & P_{53} & P_{55} \end{bmatrix} \quad (8.9)$$

In this example, terms in the second row and column are deleted and the remaining terms are compressed to form a 4×4 matrix. Terms involving 3 are changed to 4 and vice versa, corresponding to reordering the third and fourth variables.

8.2.2 State Expansion

To expand the state space, the contraction process is reversed by adding new variables to the end of the state array before the new ordering is selected. The expansion equations are given by Eqs.(8.1) and (8.2). Usually, the new states are uncorrelated with the older states so that $P_{yz} = 0$. The mean \hat{Z} and covariance P_{zz} of the new states are based upon *a priori* knowledge.

If the added states are not maintained at the end of the state vector, the above state estimate and covariance are reordered to reflect the new order. Here the Q operator simply retains and reorders all the original states, so that reordering the state vector from X to X' is expressed by Eq. (8.6). The state-error covariance is reordered to according to Eq. (8.8).

8.2.3 Simultaneous Contraction and Expansion

Situations that involve state discontinuities with no change of state dimension can be handled by combining these operations into a single step. An example of this would be a maneuver that is modeled as a discontinuity in velocity. Here Z represents the velocity components of the spacecraft. An example of changing subsystems would be a change of clock. Certain systems have multiple clocks, and one clock be substituted for another. To combine these operations into one, first add the new variables at the end, as indicated above to expand both the state estimate and covariance. Now define the operator Q as in the Eq. (8.7) and apply the operations of Eq. (8.6) and Eq. (8.8).

8.2.4 Stability Properties of the Covariance Contraction

Several properties of a contracted covariance are now proved related to numerical stability. These properties address the issue of whether the contracted covariance may be less numerically stable, i.e. contain a negative eigenvalue. The proofs show that the contracted covariance has equal or better numerical stability. *These theorems hold for any finite precision arithmetic.*

First, it is proved that if P_{XX} is positive definite (see 8.2), then the $m \times m$ submatrix P_{yy} is also positive definite. The $n \times n$ matrix P_{XX} is positive definite

if and only if $X^T P_{XX} X > 0$ for all arbitrary $n \times 1$ vectors X . Now setting the vector Z to zero, one has changed $X \rightarrow \acute{X} = \begin{bmatrix} Y \\ 0 \end{bmatrix}$. Here Y is an arbitrary $m \times 1$ vector.

$$0 < \acute{X}^T P_{XX} \acute{X} = Y^T P_{yy} Y \quad (8.10)$$

The first inequality is implied since P_{XX} is positive definite. Since Y is arbitrary, this proves that P_{yy} is positive definite. Thus, the following theorem is proved:

Theorem 1 *A contracted matrix obtained from a positive definite matrix is also positive definite.*

Although (filtered or smoothed) covariances should be positive definite, very small eigenvalues may become negative due to numerical round-off. In such cases, the spread of eigenvalues magnitudes exceeds the precision of the arithmetic. A metric that expresses the amount of precision needed to capture the range of eigenvalues of a matrix is the *condition number*. The condition number of a matrix is obtained by rescaling the variables such that the diagonal terms of the covariance are unity. Then the condition number is computed by taking the ratio of the largest eigenvalue to smallest eigenvalue of this normalized matrix. The greater the condition number, the greater precision is needed to preserve the positive definiteness of the matrix.

Next it is proved that the condition number of a contracted matrix is no greater than the original matrix. Let the largest eigenvalue for the matrix P_{XX} be defined as

$$\lambda_{\max}^x = \max \{ (X^T P_{XX} X) / (X^T X) \} \quad (8.11)$$

and the smallest be defined as

$$\lambda_{\min}^x = \min \{ (X^T P_{XX} X) / (X^T X) \} \quad (8.12)$$

The condition number of the matrix P_{XX} is defined by

$$C(P_{XX}) = \lambda_{\max}^x / \lambda_{\min}^x \quad (8.13)$$

Now let us zero out Z . Then $(\acute{X}^T P_{XX} \acute{X}) / (\acute{X}^T \acute{X}) = (Y^T P_{yy} Y) / (Y^T Y) \leq \lambda_{\max}^x$. Now, since the left side is no greater than λ_{\max}^y , it has been proven that

$$\lambda_{\max}^y \leq \lambda_{\max}^x \quad (8.14)$$

A similar proof can show that

$$\lambda_{\min}^y \geq \lambda_{\min}^x \quad (8.15)$$

This implies

$$1/\lambda_{\min}^y \leq 1/\lambda_{\min}^x \quad (8.16)$$

Multiply Eq. (8.14) by Eq. (8.16) to get

$$C(P_{yy}) = \lambda_{\max}^y / \lambda_{\min}^y \leq C(P_{XX}) = \lambda_{\max}^x / \lambda_{\min}^x \quad (8.17)$$

Theorem 2 *The condition number of a contracted matrix is no larger than the condition number of the original matrix.*

These two theorems show that contracting a covariance matrix will not cause numerical stability problems.

8.3 Smoothing Algorithms

The object of this section is to review approaches to smoothing. The extension of these methods to problems with state discontinuities will be addressed in the sequel.

Earliest smoothers consisted of fixed-interval smoothers, fixed-point smoothers, and fixed-lag smoothers ([77]). The fixed-lag smoother is not treated here; a more general multi-point smoother is formulated by McReynolds ([76]). The goal of the multi-point smoother is to produce smoothed points on an output grid that is sparse compared to the filter grid, thus reducing the amount of computations required by conventional fixed-interval smoothers.

8.3.1 Fixed-Interval Smoothing (RTS)

One of the first smoothers that gained fame is the Rauch-Tung-Striebel (RTS) [98] fixed-interval smoother. The RTS fixed-interval smoother calculates the smoothed state estimates and covariances via backward sequential operations on the input grid. By forcing the filter to operate on the output grid and sampling grid monitoring grid, the RTS fixed-interval smoother can provide a solution to the VLS problem with fixed dimension size.

Extending the Meditch notation, the RTS fixed-interval smoother, given by Eqs 6.18, 6.47, and 6.56 in reference [77] is repeated here:

$$A_X(k, k+1) = P_X(k|k) \Phi_X^T(k+1, k) P_X^{-1}(k+1|k) \quad (8.18)$$

$$\hat{X}(k|N) = \hat{X}(k|k) + A_X(k, k+1) \left[\hat{X}(k+1|N) - \hat{X}(k+1|k) \right] \quad (8.19)$$

$$P_X(k|N) = P_X(k|k) + A_X(k, k+1) \left[P_X(k+1|N) - P_X(k+1|k) \right] A_X^T(k, k+1) \quad (8.20)$$

The $n \times n$ matrix $A_X(k, k+1)$ defined by Eq. 8.18 is referred to as the "smoother gain." Originally written as $A(k)$ by Meditch, here a subscript X is added to it to denote the associated variable name. An additional argument $k+1$ is also added to emphasize that it maps smoother corrections backward in time from epoch $k+1$ to k . $A_X(k, k+1)$ can be thought of as the "smoother transition matrix." as indicated in Eq.8.19. Eq. 8.20 shows how to calculate smoothed covariances using a backward propagator.

The RTS fixed-interval smoother equations can be derived by applying the Extended Gauss-Markov Theorem [76] to the pair of vectors $\{X_k, X_{k+1}\}$. Considering the joint distribution of the errors between two vectors $\{\hat{X}_{k|k}, \hat{X}_{k+1|k}\}$,

the errors have zero mean with the covariance given by

$$\begin{bmatrix} P_X(k|k) & P_X(k, k+1|k) \\ P_X^T(k, k+1|k) & P_X(k+1|k) \end{bmatrix} \quad (8.21)$$

where $P_X(k, k+1) = P_X(k, k) \Phi^T(k+1, k)$ is the cross covariance between the two vectors $\{X_k, X_{k+1}\}$. With this information, the Extended Gauss-Markov Theorem can be used to obtain the RTS fixed-interval smoother.

8.3.2 Fixed-Point Smoothing (Carlton-Rauch)

Let us define the generalized smoother gain or a multi-stage smoother transition matrix $A_X(j, k)$, $j < k$:

$$A_X(j, k) = \prod_{j < k-1} A_X(j, j+1) \cdot A_X(j+1, j+2) \cdots A_X(k-1, k) \quad (8.22)$$

The multi-stage smoother transition matrix is obtained by multiplying together a succession of smoother transition matrices. In the fixed-point smoother this expression is sequentially determined by multiplying by the last matrix on the right.

Meditch uses B ([77], Eq. 6.61) and C ([77], Eq. 6.85) to denote this expression in different situations. Using this expression, the smoothed estimate and covariance at epoch j is given by

$$\hat{X}(j|k) = \hat{X}(j|k-1) + A_X(j, k) [\hat{X}(k|k) - \hat{X}(k|k-1)] \quad (8.23)$$

$$P_X(j|k) = P_X(j|k-1) + A_X(j, k) [P_X(k|k) - P_X(k|k-1)] A_X^T(j, k) \quad (8.24)$$

From the definition, the smoother gains must satisfy the sequential relationship (see Eq. 8.18)

$$A_X(j, k) = A_X(j, k-1) A_X(k-1, k); k = j+1, j+2, \dots \quad (8.25)$$

The second term on the right being the RTS smoother gain, (see Eq. 8.18).

8.3.3 Fixed-Point Smoother (Meditch, Anderson-Moore)

Another form of the fixed-point smoother was developed by Meditch ([77], p. 232) He credits Fraser for this, but a literature search does not seem to support this (Fraser was more concerned with fixed-interval smoothing). Meditch [77] states that his form can be developed as a corollary to the Carlton-Rauch form, but does not contain a proof. Another fixed-point smoother was developed by Anderson and Moore and is contained in reference [85], p. 170. The same equations, using the same approach, was derived later by McReynolds [76]. This development is repeated here, whereby inspection shows that this form is more numerically efficient than Meditch's presentation.

A straightforward approach to smoothers development is to formulate the fixed-pointed smoothing problem as a filtering problem where Y is the dynamic state; W is the initial conditions at epoch 0. In this application, W has a fixed epoch; however, it is treated as a quantity with a dynamic epoch like Y to avoid complexity of notation. The algorithm starts by expanding the filter at epoch 0, adding the values of Y at epoch 0 as parameters to the filter. The initial conditions of this expanded filtering problem are given by

$$P = \begin{bmatrix} P_{y0} & P_{yw0} \\ P_{yw0}^T & P_{w0} \end{bmatrix} \quad (8.26)$$

$$\hat{X} = \begin{bmatrix} \hat{y}_0 \\ \hat{w}_0 \end{bmatrix} \quad (8.27)$$

Here we set

$$\hat{w}_0 = \hat{y}_0 \quad (8.28)$$

$$P_{z0} = P_{y0} \quad (8.29)$$

$$P_{yz0} = P_{y0} \quad (8.30)$$

The transition matrix is given by

$$\Phi_X = \begin{bmatrix} \Phi_y & 0 \\ 0 & I \end{bmatrix} \quad (8.31)$$

Where I is the identify matrix. Now assuming the mean

$$\hat{X}(k|k) = \begin{bmatrix} \hat{Y}(k|k) \\ \hat{W}(k|k) \end{bmatrix} \quad (8.32)$$

and covariance matrix

$$P(k|k) = \begin{bmatrix} P_{yy}(k|k) & P_{yw}(k|k) \\ P_{yw}^T(k|k) & P_{ww}(k|k) \end{bmatrix} \quad (8.33)$$

are calculated at t_k , the predicted mean and covariance are given by

$$\hat{X}(k+1|k) = \begin{bmatrix} \hat{Y}(k+1|k) \\ \hat{W}(k+1|k) \end{bmatrix} = \begin{bmatrix} \Phi_y(k+1, k) \hat{Y}(k|k) \\ \hat{W}(k|k) \end{bmatrix} \quad (8.34)$$

and

$$P(k+1|k) = \begin{bmatrix} P_{yy}(k+1|k) & P_{yw}(k+1|k) \\ P_{yw}^T(k+1|k) & P_{ww}(k+1|k) \end{bmatrix} \quad (8.35)$$

where

$$P_{yy}(k+1|k) = \Phi_y(k+1, k) P_{yy}(k|k) \Phi_y^T(k+1, k) + Q_y(k+1, k) \quad (8.36)$$

$$P_{yw}(k+1|k) = \Phi_y(k+1, k) P_{yw}(k|k) \quad (8.37)$$

$$P_{ww}(k+1|k) = P_{ww}(k|k) \quad (8.38)$$

Here the matrix $Q_y(k+1, k)$ represents the process noise over the time interval $(k+1, k)$ for the dynamic variable Y .

Next assume the following measurement is given by

$$M(k+1) = H_X(k+1) X(k+1) + V(k+1) \quad (8.39)$$

where

$$H_X(k+1) = [H_y(k+1) \quad 0] \quad (8.40)$$

$V(k+1)$ is zero-mean white noise with covariance given by $R(k+1)$. Now we apply the conventional Kalman filter equations. First note,

$$H_X(k+1) P_{XX}(k+1|k) = [H_y(k+1) P_{yy}(k+1|k) \quad H_y(k+1) P_{yw}(k+1|k)] \quad (8.41)$$

$$H_X(k+1) P_{XX}(k+1|k) H_X^T(k+1) = H_y(k+1) P_{yy}(k+1|k) H_y^T(k+1) \quad (8.42)$$

These quantities will be needed in the following equations. The covariance of the predicted residuals at $k+1$ is given by

$$P_{mm}(k+1|k) = H_y(k+1) P_{yy}(k+1|k) H_y^T(k+1) + R(k+1) \quad (8.43)$$

The Kalman gain ([77], Eq. 5.49) is given by

$$K_X(k+1) = P_{XX}(k+1) H_X^T(k+1) P_{mm}^{-1}(k+1|k) \quad (8.44)$$

where the product of the first two terms is given by Eq. 8.41 transposed. Partitioning this equation for the Kalman gain gives

$$K_X(k+1) = \begin{bmatrix} K_y(k+1) \\ K_w(k+1) \end{bmatrix} \quad (8.45)$$

$$K_y(k+1) = P_{yy}(k+1|k) H_y^T(k+1) P_{mm}^{-1}(k+1|k) \quad (8.46)$$

$$K_w(k+1) = P_{wy}(k+1|k) H_y^T(k+1) P_{mm}^{-1}(k+1|k) \quad (8.47)$$

Note that the gain $K_w(k+1)$ does not depend on the covariance $P_{ww}(k+1|k)$ because the measurements do not depend upon W . The updated means are given by ([77], Eq. 5.48)

$$\hat{y}(k+1|k+1) = \hat{y}(k+1|k) + K_y(k+1) [M(k+1) - H_y(k+1) \hat{y}(k+1|k)] \quad (8.48)$$

$$\hat{w}(k+1|k+1) = \hat{w}(k+1|k) + K_w(k+1) [M(k+1) - H_y(k+1) \hat{y}(k+1|k)] \quad (8.49)$$

The updated covariances ([77], Eq. 5.51) are given by

$$P_{yy}(k+1|k+1) = [I - K_y(k+1) \cdot H_y(k+1)] P_{yy}(k+1|k) \quad (8.50)$$

$$P_{ww}(k+1|k+1) = P_{ww}(k+1|k) - K_w(k+1) H_y(k+1) P_{yw}(k+1|k) \quad (8.51)$$

$$P_{yw}(k+1|k+1) = [I - K_y(k+1) H_y(k+1)] P_{yw}(k+1|k) \quad (8.52)$$

To obtain the Meditch equations, the following alternative for the Kalman gain is used ([77], Eq. 5.78):

$$K(k+1) = P(k+1|k+1) H^T(k+1) \cdot R^{-1}(k+1) \quad (8.53)$$

This results in the following equations for the gains

$$K_y(k+1) = P_{yy}(k+1|k+1) H_y^T(k+1) R^{-1}(k+1) \quad (8.54)$$

$$K_w(k+1) = P_{zy}(k+1|k+1) H_y^T(k+1) R^{-1}(k+1) \quad (8.55)$$

The Fraser form is not as efficient as the first form, since these last two equations need not be calculated (one already has the gains as intermediate computations to update the covariance; see Eqs 8.46 and 8.47).

In comparing these equations to those by Meditch, it should be noted that the following matrix relationship holds between the matrix W used by Meditch ([77], p. 232) and the matrices calculated here:

$$W(0, k) = P_{wy}(k) \quad (8.56)$$

The filter equations for Y are same as the Kalman filter, with or without the presence of the states W . Thus, the solution for W does not impact Y ; note that, on the other hand, the solution for Y impacts the solution W .

If the smoothed solutions for more than one epoch, the algorithm can handle this by growing vector W by adding new states. To enforce the maximum lag constraint components of the vector W can be dropped from the filter.

The computation of the matrix P_{ww} is not usually needed in the algorithm to compute the mean. Not all components of matrix may be desired. If only the covariances of the states at the same epoch, are desired, Eq. 8.51 can be arranged so that only the computations of these components are performed.

The advantage of the Fraser fixed-point smoother over other forms (the RTS fixed-interval smoother and the Carlton-Rauch multi-stage smoother) is that the inverse of the of $n \times n$ covariance is not required, n being the dimension of the dynamic state.

8.3.4 McReynolds Multi-point Smoother

Although the Fraser form of the smoother is more efficient than other forms discussed so far, it has a problem in that the number of output epochs grow. Increasing computations tend to be proportional to the number of epochs that are within the maximum lag window and the density of data points. A strategy to reduce the number of needed solutions for filter updates is to create a parallel structure by concatenating a set of fixed-point filters.

Let k and $k+p$ be two successive epochs for which smoothed outputs are wanted. In the multi-point formulation, a fixed-point filter produces a joint solution on those grid points, namely

$$\hat{X}(k|k+p-1), \hat{X}(p+k|k+p-1) \quad (8.57)$$

Here, p is the number of measurement epochs between smoothed outputs. The fixed-point filter also produces the joint covariance of these variables, namely

$$S(k, k+p) = \begin{bmatrix} P(k|k+p-1) & P(k, k+p|k+p-1) \\ P^T(k, k+p|k+p-1) & P(k+p|k+p-1) \end{bmatrix} \quad (8.58)$$

This sequence of solutions is tied together by applying the Extended Gauss-Markov Theorem backwards in time. Assuming the smoother has performed the following updates by tying together $S(k+p, k+2p)$, $S(k+2p, k+3p)$, ...

$$\hat{X}(p+k|k+p-1) \rightarrow \hat{X}(p+k|N) \quad (8.59)$$

$$P(k+p|k+p-1) \rightarrow P(k+p|N) \quad (8.60)$$

The Extended Gauss-Markov Theorem can now be used with the previous equations to obtain the "smoother gain":

$$K(k, k+p) = P(k, k+p|k+p-1) P^{-1}(k+p|k+p-1) \quad (8.61)$$

Applying this gain, the following formula updates the estimate of that state at k :

$$\hat{X}(k|N) = \hat{X}(k|k+p-1) + K(k, k+p) \left[\hat{X}(p+k|N) - \hat{X}(p+k|k+p-1) \right] \quad (8.62)$$

Next the cross-covariance between these states at different epochs (k and $k+p$) is updated:

$$P(k, k+p|N) = K(k, k+p) P(k+p|N) \quad (8.63)$$

Finally, it updates the covariance at epoch k :

$$P(k|N) = P(k|k+p-1) + K(k, k+p) [P(k+p|N) - P(k+p|k+p-1)]^T \quad (8.64)$$

8.4 Handling Filter Discontinuities with Smoother

8.4.1 Fixed-Interval Smoothing with Discontinuities

If a discontinuity occurs in the filter, then the smoother must make an adjustment in the opposite sense because the filter and smoother operate in reverse time with each other. If the filter expands its state, the smoother must contract the state; if the filter contracts its state, then the smoother expands its state.

Consider the case where the filter expands its state at epoch k . When the backward smoother reaches the epoch k , then the smoother produces the following quantities (see Eqs. 8.1 and 8.2)

$$\hat{X}(k|N) = \begin{bmatrix} \hat{Y}(k|N) \\ \hat{Z}(k|N) \end{bmatrix} \quad (8.65)$$

$$P_X(k|N) = \begin{bmatrix} P_{yy}(k|N) & P_{yz}(k|N) \\ P_{zy}(k|N) & P_{zz}(k|N) \end{bmatrix} \quad (8.66)$$

The transition matrix can also be partitioned as follows:

$$\Phi_X = \begin{bmatrix} \Phi_{yy} & \Phi_{yz} \\ \Phi_{zy} & \Phi_{zz} \end{bmatrix} \quad (8.67)$$

To proceed backwards, only smoothed values of Y need to be produced, such that the dimension of the smoother gain is reduced. Thus one can propagate the smoother equations for the mean and covariance of Y backwards from the discontinuity using the smoother equations for the reduced state. The smoothed values of the mean and covariance has been generated at the epoch k . Then the smoother transition matrix for the reduced state is given by

$$A_y(j, j+1) = P_{yy}(j|j) \Phi_{yy}^T(j+1, j) P_{yy}^{-1}(j+1|j) \quad (8.68)$$

Next we consider the case when the filter reduces the dimension of the state. Let us assume Z is eliminated from the filter state at epoch k such that the smoother starts with the filtered state of Y at N . The backward smoother produces only the smoothed estimate and covariance of Y until it reaches the epoch k . Next we expand the smoother expands to update the joint estimate of Y and Z , according the following equations, based upon application of the Extended Gauss-Markov Theorem:

$$L_{zy}(k) = P_{zy}(k|k) P_{yy}^{-1}(k|k) \quad (8.69)$$

$$\hat{Z}(k|N) = \hat{Z}(k|k) + L_{zy}(k) \left(\hat{Y}(k|N) - \hat{Y}(k|k) \right) \quad (8.70)$$

$$P_{zy}(k|N) = L_{zy}(k) P_{yy}(k|N) \quad (8.71)$$

$$P_{zz}(k|N) = P_{zz}(k|k) + L_{zy}(k) (P_{yz}(k|N) - P_{yz}(k|k)) \quad (8.72)$$

These smoother equations parallel that of the RTS smoother, using the operator

$$A_Z^E(k) = \begin{bmatrix} I \\ L_{zy}(k) \end{bmatrix} \quad (8.73)$$

This operator applies at the time of the discontinuity, t_k the smoother adds the state vector Z to Y . The superscript E is used to emphasize that the state is being expanded by the smoother. One can also define a smoother compaction operator that deletes Z from the solution:

$$A_Z^C(k) = [I \quad 0] \quad (8.74)$$

These can be combined with the operator

$$A_Z^{EC} = \begin{bmatrix} I & 0 \\ L_{zy}(k) & 0 \end{bmatrix} \quad (8.75)$$

This operator will delete Z from the previous solution and add a new solution for Z .

If the new states are contracted or reordered by the operator Q , then this operator must be modified by multiplication on the right by Q^T , e.g.

$$A_Z^E(k) = \begin{bmatrix} Q^T \\ L_{zy}(k) Q^T \end{bmatrix} \quad (8.76)$$

$$A_Z^C(k) = [Q^T \quad 0] \quad (8.77)$$

$$A_Z^{EC} = \begin{bmatrix} Q^T & 0 \\ L_{zy}(k) Q^T & 0 \end{bmatrix} \quad (8.78)$$

8.4.2 Fixed-Point Smoothing (Carlton-Rauch) with Discontinuities

Consider the case where the filter state must be expanded. If the fixed-point is at or after the discontinuity, then the current fixed-point smoother applies without modification. Let us consider the case where the fixed-point occurs before the discontinuity. Now if the epoch of the filter is before the discontinuity, then the existing fixed-point equations apply for the contracted state Y . Now once Z is added, the fixed-point equations must be expanded. The fixed-point gain is obtained by a product of gains: the gain for the contracted set between the discontinuity and fixed-point epoch and the gain from the expanded set between the discontinuity and the current epoch. Let k be the epoch of the discontinuity, and j be the fixed epoch. Then when the epoch k is reached, the step in the fixed-point smoother is given by [77]:

$$\hat{Y}(j|k) = \hat{Y}(j|k-1) + A_y(j, k) \left(\hat{Y}(k|k) - \hat{Y}(k|k-1) \right) \quad (8.79)$$

$$P_y(j|k) = P_y(j|k-1) + A_y(j, k) (P_y(k|k) - P_y(k|k-1)) A_y^T(j, k) \quad (8.80)$$

$$A_y(j, k) = A_y(j, k-1) A_y(k-1, k) \quad (8.81)$$

where $A_y(k-1, k)$ is given in Eq. 8.18. This is the last smoother step before the filter state is expanded.

Now to continue the fixed-point smoother, the filter state must be expanded to include corrections involving the state parameters Z . Thus one initiates a fixed-point smoother that updates the augmented state and covariance to calculate: $\hat{X}(k|q)$ and $P_X(k|q)$, with $q > k$. This will produce $\hat{Y}(k|q)$ and $P_y(k|q)$, which, when substituted in the equations above, will update the estimate for Y_j and its covariance, i.e.

$$\hat{Y}(j|q) = \hat{Y}(j|q-1) + A_y(j, k) \left(\hat{Y}(k|q) - \hat{Y}(k|q-1) \right) \quad (8.82)$$

$$P_y(j|q) = P_y(j|q-1) + A_y(j, k) (P_y(k|q) - P_y(k|q-1)) A_y^T(j, k) \quad (8.83)$$

Thus, it can be seen that the smoother is accomplished in two steps: mapping to the discontinuity and then to the fixed-point epoch.

Next we consider when the state space must be decreased by the filter. Again the only problem that cannot be handled by the standard fixed-point smoother is where the smoother epoch occurs prior to the discontinuity epoch. Again the fixed-point smoother must be accomplished in 3 steps:

1. Apply the standard fixed-point-smoother to obtain to map the current epoch to the smoothed state at the discontinuity epoch
2. Expand the state estimate and covariance at the discontinuity epoch using the Extended Gauss-Markov Theorem Eqs.8.69-8.72.
3. Use the previously calculated smoother transition matrix, state, and covariance information to provide an updated smoothed estimate at the desired epochs for the expanded state.

Both cases – state contraction and state expansion – have been handled in two steps. These can be combined into one step (in spirit of the original fixed-point algorithm) by inserting a combined smoother contraction/expansion operator at k . Defined the instantaneous smoother transition as follows:

$$A_X^+(j, k) = \begin{bmatrix} A_X^-(j, k) & \text{No discontinuity at } k \\ A_X^-(j, k) A_Z^E(k) & \text{Filter subtracts } Z \text{ from } X \\ A_X^-(j, k) A_Z^E(k) & \text{Filter adds } Z \text{ to } Y \\ A_X^-(j, k) A_Z^{EC}(k) & \text{Combined operations} \end{bmatrix} \quad (8.84)$$

Here a discontinuity is inserted into the smoother gain where the superscripted signs $^+$ and $^-$ refer to before and after the epoch t_k .

8.4.3 Fixed-Point Smoothing (Fraser) with Discontinuities

The Fraser form of the fixed-point smoother can be handled as the Rauch-Carlton smoother. However, in lieu of contracting and expanding the smoother gain, the cross covariance between the dynamic state and smoothed parameters must be expanded or contracted at the epoch of discontinuity k . At the same time, the state vector might be reordered. Expressed using the McReynolds algorithm formulation:

$$P_X^+(j, k|k) = \begin{bmatrix} P_X^-(j, k|k) & \text{No discontinuity or reordering at } k \\ P_X^-(j, k|k) Q^T(k) & \text{Filter subtracts, reorders per } Q \\ P_X^-(j, k|k) A_Z^C(k) Q^T(k) & \text{Filter adds, subtracts, reorders per } Q \end{bmatrix} \quad (8.85)$$

8.4.4 Multi-Point Smoothing with Discontinuities

Let us assume that the filter discontinuity is located at an output epoch of the multi-point filter. Only two cases that need to be considered: filter expansion and filter contraction. In both cases, the problem is to join two successive fixed-point filters at a common epoch: the "earlier solution" and the "later solution." In the filter expansion case, the algorithm generates a smoothed solution for larger state, and then the algorithm simply extracts the solution of the states in common with the earlier solution. The Extended Gauss-Markov Theorem updates the earlier solution. In the filter contraction case, the same approach can be applied. The states in common to both solutions will be contained in the later solution. The smoother uses the Extended Gauss-Markov Theorem to expand the smoothed solution to include all states in the previous solution.

Here we focus on the Carlson-Rauch (CR) and Fraser (F) implementation aspect associated with discontinuities. For each algorithm, filter expansion and filter contraction are considered. Rather than be concerned with the entire VLS implementation, we shall only be concerned with specific operations required by the discontinuity. In each case, the filter needs to be expanded (states added) or contracted (states deleted), modifying the state estimate and covariance as previously discussed. Let us denote these states by Z . If states are deleted the remaining states are designed by Y . The state vector is denoted by X . If states are to be added, an *a priori* mean and covariance need to be available for the new states. If they are correlated with the old states (which is usually not the case), the cross covariance must be supplied and added to the expanded covariance in the correct position.

Carlton-Rauch Algorithm

Note that in both cases, a discontinuity operator is applied to the gains, changing the dimensions of the gain to become a non-square matrix.

Filter Expansion The smoother contraction operator must be applied to the smoother gain:

$$A_X^+(j, k) = A_X^-(j, k) A_Z^C(k) \quad (8.86)$$

where

$$A_Z^C(k) = \begin{bmatrix} I & 0 \end{bmatrix} \quad (8.87)$$

The next update of the gain is applied to $A_X^+(j, k)$.

Filter Contraction The smoother expansion operator must be applied to the smoother gain:

$$A_X^+(j, k) = A_X^-(j, k) A_Z^E(k) \quad (8.88)$$

where

$$A_Z^E(k) = \begin{bmatrix} I \\ L_{zy}(k) \end{bmatrix} \quad (8.89)$$

The next update of the gain is applied to $A_X^+(j, k)$.

Fraser-McReynolds Smoother

These operations parallel those for the Carlton-Rauch smoother, where the discontinuous operations are applied to the cross covariance matrix between the states at the smoothed epochs and the filter epoch.

Filter Expansion The smoother contraction operator must be applied to the cross covariance. The equations for the McReynolds and Fraser algorithms are:

$$P_X^+(j, k|k) = P_X^-(j, k|k) A_Z^C(k) Q^T \quad (8.90)$$

$$W^+(k) = W^-(k) A_Z^C(k) Q^T \quad (8.91)$$

where

$$A_Z^C(k) = [I \quad 0] \quad (8.92)$$

The next update of the cross-covariance is applied to $P_X^+(j, k|k)$ or $W^+(k)$.

Filter Contraction The smoother expansion operator must be applied to the smoother gain (McReynolds followed by Fraser):

$$P_X^+(j, k|k) = P_X^-(j, k|k) Q^T \quad (8.93)$$

$$W^+(k) = W^-(k) Q^T \quad (8.94)$$

The next update of the gain is applied to $P_X^+(j, k|k)$ or $W^+(k)$.

Chapter 9

Relative Orbit Errors

Herein we derive the covariance on the estimate error of the orbit difference for any two spacecraft, say i and j . Eq. 1.35 defines the true 6×1 position and velocity matrix Z (also referred to as the orbit):

$$Z = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \end{bmatrix}$$

where z is a 3×1 matrix of position components and \dot{z} is a 3×1 matrix of velocity components. Any estimate of Z is denoted with \hat{Z} , and the error in \hat{Z} is defined with:

$$\delta Z = Z - \hat{Z}$$

9.1 Simultaneous Orbit Determination

Distinguish each orbit estimated simultaneously with a superscript Z^α , $\alpha \in \{i, j, k, \dots\}$. For two simultaneous orbits define:

$$X = \begin{bmatrix} Z^i \\ Z^j \end{bmatrix}$$
$$\hat{X} = \begin{bmatrix} \hat{Z}^i \\ \hat{Z}^j \end{bmatrix}$$
$$\delta X = \begin{bmatrix} \delta Z^i \\ \delta Z^j \end{bmatrix}$$

The simultaneous state estimate error covariance provides:

$$E \left\{ (\delta X) (\delta X)^T \right\} = \begin{bmatrix} E \left\{ (\delta Z^i) (\delta Z^i)^T \right\} & E \left\{ (\delta Z^i) (\delta Z^j)^T \right\} \\ E \left\{ (\delta Z^j) (\delta Z^i)^T \right\} & E \left\{ (\delta Z^j) (\delta Z^j)^T \right\} \end{bmatrix} \quad (9.1)$$

9.2 Orbit Difference Error Covariance

Define the difference of orbits, orbit estimates, and orbit estimate errors:

$$D_{ij} = Z^i - Z^j$$

$$\hat{D}_{ij} = \hat{Z}^i - \hat{Z}^j$$

$$\delta D_{ij} = \delta Z^i - \delta Z^j$$

Then the associated unbiased covariance on δD_{ij} is found:

$$\begin{aligned} E \left\{ (\delta D_{ij}) (\delta D_{ij})^T \right\} &= E \left\{ (\delta Z^i) (\delta Z^i)^T \right\} + E \left\{ (\delta Z^j) (\delta Z^j)^T \right\} \\ &\quad - \left[E \left\{ (\delta Z^i) (\delta Z^j)^T \right\} + E \left\{ (\delta Z^j) (\delta Z^i)^T \right\} \right] \end{aligned} \quad (9.2)$$

where the terms of Eq. 9.2 are derived as 6×6 elements of Eq. 9.1.

Chapter 10

Keplerian Variables

Herein we derive the covariance on the estimated error of the orbit for Keplerian orbital elements. Eq. 1.35 defines the true 6×1 position and velocity matrix Z (also referred to as the orbit):

$$Z(t_i) = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \end{bmatrix}$$

where z is a 3×1 matrix of position components and \dot{z} is a 3×1 matrix of velocity components. Let us describe the Keplerian orbital element vector:

$$K(t_i) = \begin{bmatrix} a \\ e \\ u \\ i \\ \Omega \\ \omega \end{bmatrix} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \\ K_6 \end{bmatrix} \quad (10.1)$$

Effectively, the covariance amounts to defining the 6×6 matrix of partials

$$\Phi_K(t_i) = \frac{\partial K(t)}{\partial Z(t)} \Phi_Z(t_i) \left[\frac{\partial K(t)}{\partial Z(t)} \right]^T \quad (10.2)$$

where $\Phi_K(t_i)$ is the 6×6 Keplerian orbit-error covariance matrix and $\Phi_Z(t_i)$ is the Cartesian orbit-error covariance matrix. The technique herein is to compute

$$\frac{\partial a}{\partial z} = \frac{\partial K_1}{\partial z} = \left[\frac{2a^2}{r^3} z_1 \quad \frac{2a^2}{r^3} z_2 \quad \frac{2a^2}{r^3} z_3 \right]^T \quad (10.3)$$

$$\frac{\partial e}{\partial z} = \left[\frac{2a^2}{r^3} z_1 \quad \frac{2a^2}{r^3} z_2 \quad \frac{2a^2}{r^3} z_3 \right]^T \quad (10.4)$$

Chapter 11

Equinoctal Variables

Eq. 1.35 defines the 6×1 position and velocity matrix Z :

$$Z(t_i) = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \end{bmatrix}$$

where z is a 3×1 matrix of position components and \dot{z} is a 3×1 matrix of velocity components. Let us describe the equinoctal orbital element vector ([56]):

$$\alpha(t_i) = \begin{bmatrix} k \\ h \\ n \\ L \\ p \\ q \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{bmatrix} \quad (11.1)$$

Effectively, the covariance amounts to defining the 6×6 matrix of partials

$$\Phi_\alpha(t_i) = \frac{\partial \alpha(t)}{\partial Z(t)} \Phi_Z(t_i) \left[\frac{\partial \alpha(t)}{\partial Z(t)} \right]^T \quad (11.2)$$

where $\Phi_\alpha(t_i)$ is the 6×6 equinoctal orbit-error covariance matrix and $\Phi_Z(t_i)$ is the Cartesian orbit-error covariance matrix.

Coordinate selection can affect the behavior of estimators in the orbit determination problem ([118]). Relative to Cartesian coordinates, orbital elements provide a better representation of the distribution of a set of trajectories which represent perturbations drawn from an initial Gaussian distribution about a nominal trajectory. Furthermore, in two-body dynamics, which dominates the overall dynamics in most cases, the representation of orbital energy through the

mean motion n , and the representation of orbital angle given by the mean longitude L , results in equations of motion that are linear rather than non-linear. Specifically, the time derivative of the equinoctial elements under two-body motion is:

$$\dot{\alpha}(t_i) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ n \\ 0 \\ 0 \end{bmatrix} \quad (11.3)$$

The time derivative of the linearized dynamics is given as:

$$\dot{\alpha}(t_i) + \Delta\dot{\alpha}(t_i) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ n + \Delta n \\ 0 \\ 0 \end{bmatrix} \quad (11.4)$$

which leads to:

$$\dot{\alpha}(t_i) + \Delta\dot{\alpha}(t_i) = \frac{d}{dt} (\alpha(t_i) + \Delta\alpha(t_i)) \quad (11.5)$$

Because the two-body dynamics are linear when expressed in equinoctial coordinates, the linearization of the dynamics becomes an exact representation of motion about the nominal trajectory and a Gaussian distribution will therefore propagate exactly as a Gaussian distribution for all time. In contrast, the linearized two-body dynamics in Cartesian coordinates are an approximation valid only in a local region about the nominal.

The connection to orbit estimation comes in the update of the non-linear state (*e.g.*, Eqs. 4.8 through 4.15). The state correction is computed under the assumption that the updated non-linear state will behave in a manner predicted by the linearized dynamics. This assumption is most valid for cases where the linearized dynamics better represent the non-linear dynamics. The significance of the improvement will be determined by size of the update. When updates are small, as is the case for the majority of orbit determination, either representation seems adequate. In cases more representative of low resolution tracking with low data density, equinoctial variables are to be preferred over Cartesian coordinates. Mathematical developments, discussion, and examples may be found in ([118]).

Chapter 12

Time Grids

All time grid times are defined in units of TDT (Terrestrial Dynamical Time).

The collection of measurement time tags defines the Measurement Time Grid (MTG). In general, the MTG is non-uniform with time. The user should have little or no control over the MTG for real data. The general rule for optimal OD: The more data the better. Although data thinning may be useful if real data is so time-dense as to slow throughput to below real-time, there is no accuracy penalty for non-uniform data density – as is the case for least squares OD. Also, data thinning must observe the dynamic threshold for information content. The dynamic threshold value is derived by testing, both with simulated data and real data. For LEO, one does not want to thin data to below a one-half minute in time step. The Filter Measurement Update is executed on the MTG only. The Filter Time Update is also executed on the MTG.

The Display Time Grid (DTG) is a uniform time grid whose density is specified by the user (with constraints). The real-time display of filter output data, as well as display data sent to disk files, is calculated and presented on the DTG. The Filter Time Update is executed on the DTG.

The Process-noise Time Grid (PTG) is an intimate dynamical mechanism used to aid in the integration of acceleration modeling errors into velocity and position errors in the sequential filter. Air-drag acceleration errors and solar pressure acceleration errors are integrated with a time lag of one unit. This unit is defined by the time density for the PTG: The time step needs to be sufficiently small to render the time lag negligible. The Filter Time Update is executed on the PTG.

The filter Restart-file Time Grid (RTG) is a uniform time grid whose time-points are a subset of the DTG. The subset density is specified by the user.

The Union Time Grid (UTG) is the union of the MTG, DTG, and PTG. It necessarily contains the RTG. The sequential filter runs discretely on the UTG.

The filter logic design, and CPU expense, for so many time grids is a nuisance. In the past it has proved convenient to require: $PTG = DTG$. This works well as long as we remember that the PTG time step must be sufficiently small. Accordingly, in the sequel, we shall discard reference to the PTG, but we shall

remember that $DTG = PTG$.

Part II

Stochastic Sequences

Chapter 13

Stochastic Sequences for OOD

Optimal orbit determination must address at least five classes of physical modeling errors:

- Errors due to electronic noise from resistance in the circuits of clocks, receivers, transmitters, and transponders
- Radio signal phase delay due to unmodeled cable lengths, antenna electromagnetic phase centers, Faraday rotation
- Tropospheric and ionospheric effects on the propagation of radio signals
- Errors due to our inability to perfectly model natural external physical forces on the spacecraft (gravity, air-drag, solar photon pressure, Earth IR albedo)
- Errors due to our inability to model forces with internal spacecraft sources (thrust, outgassing, thermal radiation)

Physical modeling errors are stochastic processes, and we sample these processes with stochastic sequences.

13.1 A Scalar Exponential Gauss-Markov Sequence

Let $x = x(t_k)$ denote a dynamic scalar random variable that satisfies the equation:

$$x(t_{k+1}) = \Phi(t_{k+1}, t_k) x(t_k) + \sqrt{1 - \Phi^2(t_{k+1}, t_k)} w(t_k), \quad k \in \{0, 1, 2, \dots\} \quad (13.1)$$

where $w(t)$ is a Gaussian white random variable with mean zero and constant variance σ_w^2 , and where:

$$x(t_0) = w(t_0) \quad (13.2)$$

$$\Phi(t_{k+1}, t_k) = e^{\alpha|t_{k+1}-t_k|} \quad (13.3)$$

$$\text{constant } \alpha < 0 \quad (13.4)$$

Let $E\{\cdot\}$ denote the linear expectation operator, and define:

$$\sigma_w^2 = E\{w^2(t_0)\} \quad (13.5)$$

Then with Eq. 13.2:

$$E\{x^2(t_0)\} = \sigma_w^2 \quad (13.6)$$

It is demonstrated below that:

$$E\{x^2(t_k)\} = \sigma_w^2, \text{ for each } k, \quad (13.7)$$

that $x(t_k)$ has a Gaussian distribution, and that $x(t_k)$ is Markov; *i.e.*, that the value of $x(t_k)$ is completely specified without looking backwards in time – without referring to $x(t_b)$ where $t_b < t_k$. Since the variance $E\{x^2(t_k)\}$ has the same value for any t_k , then $x(t_k)$ is said to be a *stationary* sequence [49]. But note that $x(t_k)$ is not constant with time (Eq. 13.1).

13.1.1 Deterministic Transitivity with Time

Consider the change from $x(t_k)$ to $x(t_{k+1})$ for:

$$x(t_{k+1}) = \Phi(t_{k+1}, t_k) x(t_k), \quad k \in \{0, 1, 2, \dots\} \quad (13.8)$$

Forward Motion is Transitive

If $t_{k-1} \leq t_k \leq t_{k+1}$, then forward linear transitivity:

$$\Phi(t_{k+1}, t_k) \Phi(t_k, t_{k-1}) = \Phi(t_{k+1}, t_{k-1}) \quad (13.9)$$

is easily demonstrated with:

$$e^{\alpha|t_{k+1}-t_k|} e^{\alpha|t_k-t_{k-1}|} = e^{\alpha(t_{k+1}-t_k)} e^{\alpha(t_k-t_{k-1})} = e^{\alpha|t_{k+1}-t_{k-1}|} \quad (13.10)$$

Backward Motion is Transitive

If $t_{k-1} \geq t_k \geq t_{k+1}$, then backward linear transitivity:

$$\Phi(t_{k-1}, t_k) \Phi(t_k, t_{k+1}) = \Phi(t_{k-1}, t_{k+1}) \quad (13.11)$$

is easily demonstrated with:

$$e^{\alpha|t_{k-1}-t_k|} e^{\alpha|t_k-t_{k+1}|} = e^{\alpha(t_{k-1}-t_k)} e^{\alpha(t_k-t_{k+1})} = e^{\alpha|t_{k-1}-t_{k+1}|} \quad (13.12)$$

Mixed Motion is Non-Transitive

Begin with t_{k-1} , go forward to t_k , then go back to t_{k+1} , where $t_{k+1} = t_{k-1} \leq t_k$. Then:

$$\Phi(t_{k+1}, t_k) \Phi(t_k, t_{k-1}) \neq \Phi(t_{k+1}, t_{k-1}) \quad (13.13)$$

because:

$$e^{\alpha|t_{k+1}-t_k|} e^{\alpha|t_k-t_{k-1}|} = e^{2\alpha(t_k-t_{k-1})} \quad (13.14)$$

Generally

Generally, $\Phi(t_{k+1}, t_k)$ is non-transitive because:

$$\Phi(t_k, t_{k+1}) \neq [\Phi(t_{k+1}, t_k)]^{-1} \quad (13.15)$$

that is, because:

$$e^{\alpha|t_{k+1}-t_k|} \neq e^{-\alpha|t_k-t_{k+1}|} \quad (13.16)$$

All Motion with Time

All motion with time, whether forward or backward, drives the deterministic variable $x(t)$ toward zero according to Eqs. 13.8 and 13.3, and the value of α .

13.1.2 Stationary Variance

For $k = 0$, Eq. 13.1 becomes:

$$x(t_1) = \Phi(t_1, t_0) x(t_0) + \sqrt{1 - \Phi^2(t_1, t_0)} w(t_0) \quad (13.17)$$

Then use this and Eq. 13.6 to get:

$$E\{x^2(t_1)\} = (\Phi(t_1, t_0))^2 E\{x^2(t_0)\} + (1 - \Phi^2(t_1, t_0)) E\{w^2(t_0)\} = \sigma_w^2 \quad (13.18)$$

because:

$$E \{x(t_0) w(t_0)\} = 0 \quad (13.19)$$

Eq. 13.7 follows by induction on the integers $k \in \{0, 1, 2, \dots\}$.

13.1.3 Propagation Time Extrema

The stochastic sequence defined by Eq. 13.1 depends on $|t_{k+1} - t_k|$ as follows.

Identity

$$(|t_{k+1} - t_k| = 0) \implies (\Phi(t_{k+1}, t_k) = 1), \text{ and } (x(t_{k+1}) = x(t_k)) \quad (13.20)$$

As $|t_{k+1} - t_k|$ is decreased to zero, the random variable $x(t)$ is driven to perfect serial correlation according to Eqs. 13.1 and 13.20.

White Noise

$$(|t_{k+1} - t_k| = \infty) \implies (\Phi(t_{k+1}, t_k) = 0), \text{ and } (x(t_{k+1}) = w(t_k)) \quad (13.21)$$

As $|t_{k+1} - t_k|$ is increased to infinity, the random variable $x(t)$ is driven toward white noise $w(t)$ according to Eqs. 13.1 and 13.21.

13.1.4 Input Control

It is convenient to set the value for α by choosing a constant value of $\tau = |t_{j+1} - t_j|$ associated with exponential half-life on the transition function Φ :

$$(e^{\alpha\tau} = 0.5) \implies (\alpha = (\ln 0.5) / \tau) \quad (13.22)$$

Force Serial Correlation

$$(\tau = \infty) \implies (\alpha = 0) \implies (\Phi(t_{k+1}, t_k) = 1), \text{ and } (x(t_{k+1}) = x(t_k)) \quad (13.23)$$

The user forces the sequence in x to be more correlated by setting τ larger.

Force White Noise

$$(\tau = 0) \implies (\alpha = -\infty) \implies (\Phi(t_{k+1}, t_k) = 0), \text{ and } (x(t_{k+1}) = w(t_k)) \quad (13.24)$$

The user forces the sequence in x to be less correlated by setting τ smaller.

Specify σ_w

The user specifies the sigma σ_w on $w(t_0)$. From Eq. 13.21 one sees that σ_w controls the variance on $x(t)$ for long propagation times. Practically, one matches σ_w to the physical sigma on $x(t)$, in the absence of measurement information.

13.1.5 Estimation

Eq. 13.1 defines a truth model for the unknown variable $x(t_k)$. Measurements of $x(t_k)$ are required in order to estimate $x(t_k)$. Given measurement values y_k of $x(t_k)$ at times t_k , and given the linear measurement model:

$$y_k = H_k x_k + \nu_k, \quad k \in \{1, 2, \dots\} \quad (13.25)$$

where:

$$H_k = H(t_k) \quad (13.26)$$

$$x_k = x(t_k) \quad (13.27)$$

$$\nu_k = \nu(t_k) \quad (13.28)$$

and where ν_k is a white Gaussian sequence, the fundamental theorem of estimation can be applied. Then at time t_k :

$$\hat{x}_{k|k} = E\{x_k | y_k\} \quad (13.29)$$

according to Sherman's Theorem. Define the error $\delta \hat{x}_{k|k}$ in $\hat{x}_{k|k}$ with:

$$\delta x_{k|k} = x_k - \hat{x}_{k|k} \quad (13.30)$$

and define the error variance $P_{k|k}$ on $\hat{x}_{k|k}$ with:

$$P_{k|k} = E\left\{(\delta x_{k|k})^2\right\} \quad (13.31)$$

Propagation of the Estimate $\hat{x}_{k|k}$

Given the next measurement at time t_{k+1} , apply the conditional expectation operator to Eq. 13.1 to get:

$$E\{x_{k+1} | y_k\} = \Phi_{k+1,k} E\{x_k | y_k\} + \sqrt{1 - \Phi_{k+1,k}^2} E\{w_{k+1} | y_k\} \quad (13.32)$$

But white noise is unobservable:

$$E\{w_{k+1} | y_k\} = E\{w_{k+1}\} = 0 \quad (13.33)$$

Then Eq. 13.32 becomes:

$$\hat{x}_{k+1|k} = \Phi_{k+1,k} \hat{x}_{k|k} \quad (13.34)$$

again using Sherman's Theorem.

Propagation of the Estimate Error $\delta x_{k|k}$

Define the error $\delta x_{k+1|k}$ in the propagated estimate $\hat{x}_{k+1|k}$ with:

$$\delta x_{k+1|k} = x_{k+1} - \hat{x}_{k+1|k} \quad (13.35)$$

Invoke Sherman's Theorem to derive the optimal estimate $\delta \hat{x}_{k+1|k}$ of the unknown propagated error $\delta x_{k+1|k}$:

$$\delta \hat{x}_{k+1|k} = E \{ \delta x_{k+1|k} | y_k \} \quad (13.36)$$

and define the propagated error variance $P_{k+1|k}$ with:

$$P_{k+1|k} = E \left\{ (\delta x_{k+1|k})^2 \right\} \quad (13.37)$$

Insert Eqs. 13.1 and 13.34 into Eq. 13.35 to get:

$$\delta x_{k+1|k} = \Phi_{k+1,k} \delta x_{k|k} + \sqrt{1 - \Phi_{k+1,k}^2} w_k \quad (13.38)$$

Insert Eq. 13.38 into Eq. 13.37 to get:

$$P_{k+1|k} = \Phi_{k+1,k}^2 P_{k|k} + (1 - \Phi_{k+1,k}^2) \sigma_w^2 \quad (13.39)$$

Invoke Eq. 13.36 to apply the conditional expectation operator to Eq. 13.38:

$$\delta \hat{x}_{k+1|k} = \Phi_{k+1,k} \delta \hat{x}_{k|k} \quad (13.40)$$

13.1.6 Stationarity

The scalar parameter sequence defined by Eq. 13.1 is only a parameter propagation sequence. It is independent of filter measurement-updates, and is shown to be stationary, according to Eq. 13.18. But the scalar parameter *error* sequence defined by Eq. 13.38 is non-stationary, according to Eq. 13.39. Stationarity is destroyed by the filter measurement-update that reduces the variance at measurement times.

13.2 The Vasicek Stochastic Sequence

The Vasicek (Wah-SI-check) stochastic sequence - an exponential Gauss-Markov sequence - enables both short-run and long-run time-varying bias estimation with a single stochastic sequence. The Vasicek sequence can be viewed as a significant extension to the Gauss-Markov sequence; it cannot be reduced to that sequence exactly, but it can produce an equivalent model numerically. The Vasicek sequences has the same transition-correlation function and the same square-root factor in the process noise term as the Gauss-Markov sequence presented previously.

The Vasicek sequence was designed by Vasicek for use in finance to simultaneously model the short-run interest rate level and the long-run interest rate

level, and to simultaneously model the short-run price of a bond and the long-run price of a bond ([28] page 110 Equation 3.46). For $k \in \{0, 1, 2, \dots, n\}$, let $V_k = V(t_k)$ denote the Vasicek random variable at time t_k :

$$V_{k+1} = \Phi_{k+1,k} V_k + [1 - \Phi_{k+1,k}] b + \sqrt{1 - \Phi_{k+1,k}^2} \left[\frac{\sigma Z_{k+1}}{\sqrt{2a}} \right] \quad (13.41)$$

where a and σ are positive constants, where b is a constant (positive, zero, or negative), where Z_1, Z_2, \dots, Z_n are independent draws from the unitless Standard Normal Distribution $N(0, 1)$, and where the exponential transition-correlation function $\Phi_{k+1,k}$ is defined by

$$\Phi_{k+1,k} = \exp(-a[t_{k+1} - t_k]) \quad (13.42)$$

$$t_k \leq t_{k+1}$$

For $(t_{k+1} - t_k) \rightarrow 0$

$$\lim_{(t_{k+1}-t_k) \rightarrow 0} \Phi_{k+1,k} = 1 \quad (13.43)$$

For $(t_{k+1} - t_k) \rightarrow \infty$

$$\lim_{(t_{k+1}-t_k) \rightarrow \infty} \Phi_{k+1,k} = \lim_{(t_{k+1}-t_k) \rightarrow \infty} \exp(-a[t_{k+1} - t_k]) = 0 \quad (13.44)$$

Otherwise

$$0 < \Phi_{k+1,k} < 1$$

The exponential half-life, denoted $\tau_{1/2}$, specifies a value for a . Insert $\Phi_{k+1,k} = 0.5$ and $[t_{k+1} - t_k] = \tau_{1/2}$ into Equation 13.42:

$$\exp(-a\tau_{1/2}) = 0.5$$

or

$$a = -\frac{\ln 0.5}{\tau_{1/2}} \quad (13.45)$$

where $\ln 0.5 \cong -0.693147181$. A state-space form of Equation 13.41 is

$$X_{k+1} = \bar{\Phi}_{k+1|k} X_k + \Gamma_{k+1,k} U_{k+1} \quad (13.46)$$

where

$$X_{k+1} \triangleq \begin{bmatrix} V_{k+1} \\ b \end{bmatrix} \quad (13.47)$$

$$X_k \triangleq \begin{bmatrix} V_k \\ b \end{bmatrix} \quad (13.48)$$

$$\bar{\Phi}_{k+1|k} \triangleq \begin{bmatrix} \Phi_{k+1,k} & 1 - \Phi_{k+1,k} \\ 0 & 1 \end{bmatrix} \quad (13.49)$$

$$\Gamma_{k+1,k} \triangleq \begin{bmatrix} \sqrt{1 - \Phi_{k+1,k}^2} & 0 \\ 0 & 0 \end{bmatrix} \quad (13.50)$$

$$U_{k+1} \triangleq \begin{bmatrix} \sigma Z_{k+1} / \sqrt{2a} \\ 0 \end{bmatrix} \quad (13.51)$$

13.2.1 Transition Matrix $\bar{\Phi}$

Identity

$$\bar{\Phi}_{i,i} = \begin{bmatrix} \Phi_{i,i} & 1 - \Phi_{i,i} \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \exp(-a[t_i - t_i]) & 1 - \exp(-a[t_i - t_i]) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 - 1 \\ 0 & 1 \end{bmatrix}$$

Transitivity

Since

$$\exp(-a[t_k - t_j]) \exp(-a[t_j - t_i]) = \exp(-a[t_k - t_i])$$

then

$$\Phi_{k,j} \Phi_{j,i} = \Phi_{k,i}$$

and

$$\begin{aligned} \begin{bmatrix} \Phi_{k,j} & 1 - \Phi_{k,j} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \Phi_{j,i} & 1 - \Phi_{j,i} \\ 0 & 1 \end{bmatrix} &= \begin{bmatrix} \Phi_{k,j} \Phi_{j,i} & \Phi_{k,j} [1 - \Phi_{j,i}] + [1 - \Phi_{k,j}] \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \Phi_{k,j} \Phi_{j,i} & 1 - \Phi_{k,j} \Phi_{j,i} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \Phi_{k,i} & 1 - \Phi_{k,i} \\ 0 & 1 \end{bmatrix} \end{aligned}$$

That is

$$\bar{\Phi}_{k,j} \bar{\Phi}_{j,i} = \bar{\Phi}_{k,i}$$

13.2.2 Units

The positive constant a has units inverse time (time^{-1}). Objects V_k , V_{k+1} , b , and $(\sigma/\sqrt{2a})$ have the same units, defined by the one who defines $V(t_k)$. The positive constant σ is called the *diffusion coefficient*, and has units of V_{k+1} divided by units of $\text{time}^{1/2}$.

13.2.3 Volatility

The volatility of V_{k+1} is driven by the constant value for σ , the volatility coefficient. Increasing the σ value implies more volatility.

13.2.4 Drift

The term $b[1 - \Phi_{k+1,k}]$ of Equation 13.41 generates a drift in V_{k+1} toward b with rate $\Phi_{k+1,k}$, where for our applications the constant b may be positive, negative, or zero.

13.2.5 Mean

$$E\{V_{k+1}\} = \Phi_{k+1,k}E\{V_k\} + b[1 - \Phi_{k+1,k}] \quad (13.52)$$

because

$$E\{Z_{k+1}\} = 0$$

Take limit $(t_{k+1} - t_k) \rightarrow \infty$

$$\lim_{(t_{k+1}-t_k) \rightarrow \infty} E\{V_{k+1}\} = b \quad (13.53)$$

Thus the expectation of V_{k+1} always reverts to b on the infinite time horizon with rate $\Phi_{k+1,k}$.

13.2.6 Second Moments

About Origin

The second moment about the origin is:

$$\begin{aligned} E\{V_{k+1}^2\} &= E\left\{\left[\Phi_{k+1,k}V_k + b[1 - \Phi_{k+1,k}] + \frac{\sigma}{\sqrt{2a}}\sqrt{1 - \Phi_{k+1,k}^2}Z_{k+1}\right]^2\right\} \\ E\{V_{k+1}^2\} &= E\left\{[\Phi_{k+1,k}V_k]^2\right\} + E\left\{[b[1 - \Phi_{k+1,k}]]^2\right\} + \\ &E\left\{\left[\frac{\sigma}{\sqrt{2a}}\sqrt{1 - \Phi_{k+1,k}^2}Z_{k+1}\right]^2\right\} + \end{aligned}$$

$$\begin{aligned}
& E \left\{ [\Phi_{k+1,k} V_k] \left[\frac{\sigma}{\sqrt{2a}} \sqrt{1 - \Phi_{k+1,k}^2} Z_{k+1} \right] \right\} + \\
& E \{ [\Phi_{k+1,k} V_k] [b[1 - \Phi_{k+1,k}]] \} + \\
& E \left\{ [b[1 - \Phi_{k+1,k}]] \left[\frac{\sigma}{\sqrt{2a}} \sqrt{1 - \Phi_{k+1,k}^2} Z_{k+1} \right] \right\}
\end{aligned}$$

So

$$\begin{aligned}
E \{ V_{k+1}^2 \} &= \Phi_{k+1,k}^2 E \{ V_k^2 \} + [b[1 - \Phi_{k+1,k}]]^2 + \frac{\sigma^2}{2a} [1 - \Phi_{k+1,k}^2]^2 E \{ Z_{k+1}^2 \} \\
&+ \Phi_{k+1,k} [b[1 - \Phi_{k+1,k}]] E \{ V_k \}
\end{aligned}$$

$$\begin{aligned}
E \{ V_{k+1}^2 \} &= \Phi_{k+1,k}^2 E \{ V_k^2 \} + \frac{\sigma^2}{2a} [1 - \Phi_{k+1,k}^2]^2 \\
&+ \Phi_{k+1,k} [b[1 - \Phi_{k+1,k}]] E \{ V_k \} + [b[1 - \Phi_{k+1,k}]]^2
\end{aligned}$$

$$\begin{aligned}
E \{ V_{k+1}^2 \} &= \Phi_{k+1,k}^2 E \{ V_k^2 \} + \frac{\sigma^2}{2a} [1 - \Phi_{k+1,k}^2]^2 \\
&+ [b[1 - \Phi_{k+1,k}]] [\Phi_{k+1,k} E \{ V_k \} + [b[1 - \Phi_{k+1,k}]]]
\end{aligned}$$

because

$$E \{ V_k Z_{k+1} \} = 0$$

$$E \{ Z_{k+1} \} = 0$$

$$E \{ Z_{k+1}^2 \} = 1$$

Thus for second moment about the origin

$$\begin{aligned}
E \{ V_{k+1}^2 \} &= \Phi_{k+1,k}^2 E \{ V_k^2 \} + \frac{\sigma^2}{2a} [1 - \Phi_{k+1,k}^2]^2 \\
&+ [b[1 - \Phi_{k+1,k}]] [\Phi_{k+1,k} E \{ V_k \} + [b[1 - \Phi_{k+1,k}]]]
\end{aligned} \tag{13.54}$$

Take limit $(t_{k+1} - t_k) \rightarrow \infty$

$$\lim_{(t_{k+1} - t_k) \rightarrow \infty} E \{ V_{k+1}^2 \} = \frac{\sigma^2}{2a} + b^2 \tag{13.55}$$

About Mean

For the second moment about the mean (variance), define

$$\text{VAR} = E \left\{ (V_{k+1} - E \{V_{k+1}\})^2 \right\} \quad (13.56)$$

Then

$$\begin{aligned} E \left\{ (V_{k+1} - E \{V_{k+1}\})^2 \right\} &= \left[E \{V_{k+1}^2\} + (E \{V_{k+1}\})^2 - 2[E \{E \{V_{k+1}\} V_{k+1}\}] \right] \\ &= \left[E \{V_{k+1}^2\} + (E \{V_{k+1}\})^2 - 2(E \{V_{k+1}\})^2 \right] \\ &= E \{V_{k+1}^2\} - (E \{V_{k+1}\})^2 \end{aligned}$$

That is

$$E \left\{ (V_{k+1} - E \{V_{k+1}\})^2 \right\} = E \{V_{k+1}^2\} - (E \{V_{k+1}\})^2 \quad (13.57)$$

Insert 13.54 into 13.57

$$\begin{aligned} E \left\{ (V_{k+1} - E \{V_{k+1}\})^2 \right\} &= \Phi_{k+1,k}^2 E \{V_k^2\} + \frac{\sigma^2}{2a} [1 - \Phi_{k+1,k}^2]^2 \\ &\quad + [b[1 - \Phi_{k+1,k}]] [\Phi_{k+1,k} E \{V_k\} + [b[1 - \Phi_{k+1,k}]]] - (E \{V_{k+1}\})^2 \end{aligned} \quad (13.58)$$

Take limit $(t_{k+1} - t_k) \rightarrow \infty$

$$\lim_{(t_{k+1}-t_k) \rightarrow \infty} E \left\{ (V_{k+1} - E \{V_{k+1}\})^2 \right\} = \frac{\sigma^2}{2a} \quad (13.59)$$

because b^2 subtracts out.

13.2.7 Summary

The mean $E \{V_{k+1}\} \rightarrow b$, and $\text{VAR}(V(t_k)) \rightarrow \sigma^2/(2a)$ as $t_{k+1} - t_k \rightarrow \infty$. Equation 13.41 has a short-term behaviour characterized by V_k , and a long-term behaviour characterized by b .

13.2.8 Example

If $\tau_{1/2} = 110$ min, then $a = 0.006301338 \text{ min}^{-1}$. Figure 13.1 graphs functions $f_1(t) \triangleq \Phi(t, 0) = \exp(-0.006301338 t)$ and $f_2(t) \triangleq 0.5$, with the point $(\tau_{1/2}, f(\tau_{1/2})) = (110 \text{ minutes}, 0.5)$ defined by intersection of functions $f_1(t)$ and $f_2(t)$.

Figures 13.2 and 13.3 employ half-life $\tau_{1/2} = 110$ minutes, and present Vasicek sequence results on the interval $[0, 220]$ minutes, twice the half-life $\tau_{1/2}$.

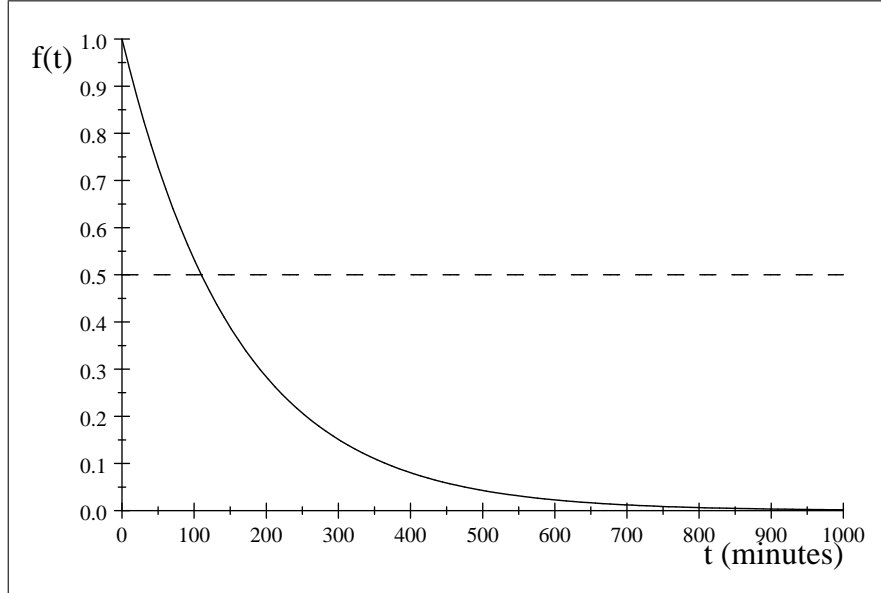


Figure 13.1: Exponential Transition-Correlation Function and Half-Life Function

Figures 13.2 and 13.3 present results using Equation 13.41. Here V_k is a one-dimensional velocity with units of meters per minute (m/min) with Vasicek bias $b = 0$ m/min for Figure 13.2, and $b = 15$ m/min for Figure 13.3. For the horizontal lines $\pm 2\sigma$ on each figure, $\sigma = \sqrt{\text{VAR}}$, where VAR is the variance about the mean according to Equation 13.56. Each Vasicek trajectory has half-life 110 minutes and length 220 minutes; i.e., length is twice half-life.

For Figure 13.3, note that at $t = 220$ minutes the visual ensemble mean has not reached the bias $b = 15$ m/min. This is explained graphically by Figure 13.1. The bias $b = 15$ m/min cannot be reached exactly on any finite time horizon.

13.2.9 Sequential Estimation

Let $V(t)$ denote a true unknown Vasicek function, let b denote a true unknown Vasicek parameter, and let $\hat{V}_k = \hat{V}(t_k)$ and $\hat{b}_k = \hat{b}(t_k)$ denote known optimal estimates of V_k and b at time t_k due to processing measurements y_k sequentially, for $k \in \{0, 1, 2, \dots\}$. I assume that V_k and b are observable to y_{k+1} . That is, partial derivatives $\partial y_{k+1} / \partial V_k$ and $\partial b / \partial V_k$ are non-zero for some non-empty subset of measurements y_{k+1} . Inspection of Equation 13.41 reveals that correlation between errors in the estimates of V_k and b is reduced by existence of the factor $\Phi_{k+1,k}$ in the term $\Phi_{k+1,k} V_k$. Thus estimates of V_k and b are separable.

Let $\hat{V}_{k|j}$ and $\hat{b}_{k|j}$ denote estimates of V_k and b at epoch t_k due to processing the last measurement y_j with time-tag t_j . Define unknown estimation errors

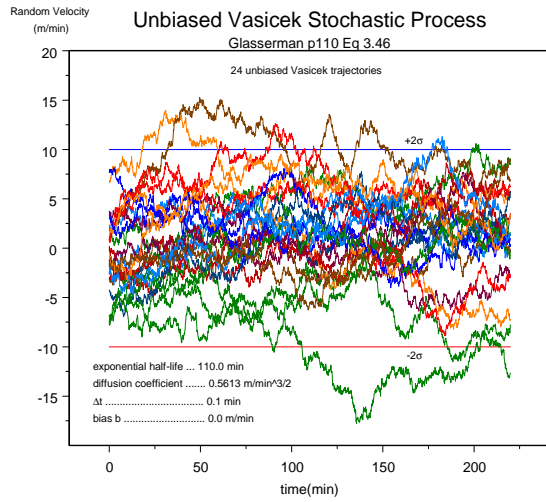


Figure 13.2: 24 Unbiased Vasicek Sequences

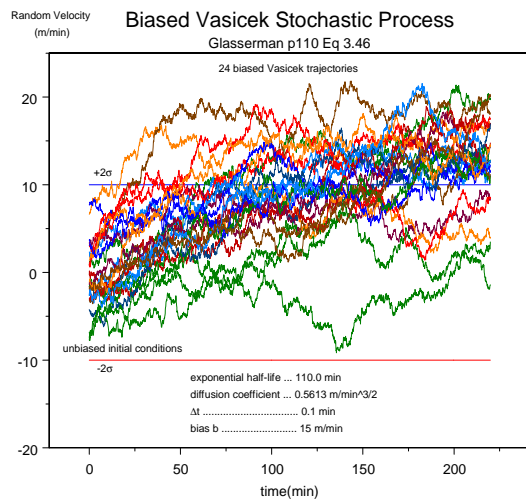


Figure 13.3: 24 Biased Vasicek Sequences

$\delta\hat{V}_{k|j}$ and $\delta\hat{b}_{k|j}$ in estimates $\hat{V}_{k|j}$ and $\hat{b}_{k|j}$ with

$$\delta\hat{V}_{k|j} = V_k - \hat{V}_{k|j} \quad (13.60)$$

$$\delta\hat{b}_{k|j} = b - \hat{b}_{k|j} \quad (13.61)$$

If $t_j < t_k$, then $\delta\hat{V}_{k|j}$ and $\delta\hat{b}_{k|j}$ are propagated errors (from t_j to t_k), where y_j was the last measurement processed. If $t_j = t_k$, then $\delta\hat{V}_{k|j} = \delta\hat{V}_{k|k}$ and $\delta\hat{b}_{k|j} = \delta\hat{b}_{k|k}$ are errors in estimates $\hat{V}_{k|k}$ and $\hat{b}_{k|k}$ due to having processed measurement y_k .

$$\delta\hat{V}_{k|k} = V_k - \hat{V}_{k|k} \quad (13.62)$$

$$\delta\hat{b}_{k|k} = b - \hat{b}_{k|k} \quad (13.63)$$

From Sherman's Theorem the optimal estimate of V_{k+1} given y_k , denoted $\hat{V}_{k+1|k}$ is

$$\hat{V}_{k+1|k} = E\{V_{k+1}|y_k\} \quad (13.64)$$

Insert Equation 13.41 into Equation 13.64 to get the filter propagation equation

$$\hat{V}_{k+1|k} = \Phi_{k+1,k}\hat{V}_{k|k} + [1 - \Phi_{k+1,k}]\hat{b}_{k|k} \quad (13.65)$$

because $E\{Z_{k+1}|y_k\} = E\{Z_{k+1}\} = 0$. Note that

$$\hat{X}_{k+1|k} = \begin{bmatrix} \hat{V}_{k+1|k} \\ \hat{b}_{k+1|k} \end{bmatrix} \quad (13.66)$$

State estimates $\hat{V}_{k|k}$ and $\hat{b}_{k|k}$ are propagated according to Equation 13.65, or

$$\hat{X}_{k+1|k} = \bar{\Phi}_{k+1|k}\hat{X}_{k|k} \quad (13.67)$$

where

$$\bar{\Phi}_{k+1|k} \triangleq \begin{bmatrix} \Phi_{k+1,k} & 1 - \Phi_{k+1,k} \\ 0 & 1 \end{bmatrix} \quad (13.68)$$

Propagation errors

$$\delta\hat{b}_{k+1|k} = b - \hat{b}_{k+1|k}$$

$$\delta\hat{V}_{k+1|k} = V_{k+1} - \hat{V}_{k+1|k}$$

$$\begin{aligned} &= \left[\Phi_{k+1,k}V_k + [1 - \Phi_{k+1,k}]b + \frac{\sigma}{\sqrt{2a}}\sqrt{1 - \Phi_{k+1,k}^2}Z_{k+1} \right] \\ &\quad - \left[\Phi_{k+1,k}\hat{V}_{k|k} + [1 - \Phi_{k+1,k}]\hat{b}_{k|k} \right] \end{aligned}$$

$$= \Phi_{k+1,k} [V_k - \hat{V}_{k|k}] + [1 - \Phi_{k+1,k}] [b - \hat{b}_{k|k}] + \frac{\sigma}{\sqrt{2a}} \sqrt{1 - \Phi_{k+1,k}^2} Z_{k+1}$$

So propagation error $\delta\hat{V}_{k+1|k}$ in $\hat{V}_{k+1|k}$ is found to be

$$\delta\hat{V}_{k+1|k} = \Phi_{k+1,k} \delta\hat{V}_{k|k} + [1 - \Phi_{k+1,k}] \delta\hat{b}_{k|k} + \frac{\sigma}{\sqrt{2a}} \sqrt{1 - \Phi_{k+1,k}^2} Z_{k+1} \quad (13.69)$$

or in state space form

$$\delta\hat{X}_{k+1|k} = \bar{\Phi}_{k+1|k} \delta\hat{X}_{k|k} + \Gamma_{k+1,k} U_{k+1} \quad (13.70)$$

where

$$\delta\hat{X}_{k+1|k} \triangleq \begin{bmatrix} \delta\hat{V}_{k+1|k} \\ \delta\hat{b}_{k+1|k} \end{bmatrix} \quad (13.71)$$

$$\delta\hat{X}_{k|k} \triangleq \begin{bmatrix} \delta\hat{V}_{k|k} \\ \delta\hat{b}_{k|k} \end{bmatrix} \quad (13.72)$$

$$\Gamma_{k+1,k} \triangleq \begin{bmatrix} \sqrt{1 - \Phi_{k+1,k}^2} & 0 \\ 0 & 0 \end{bmatrix} \quad (13.73)$$

$$U_{k+1} \triangleq \begin{bmatrix} \sigma Z_{k+1} / \sqrt{2a} \\ 0 \end{bmatrix} \quad (13.74)$$

State estimate error covariance propagation is defined by

$$P_{k+1|k} = E \left\{ \delta\hat{X}_{k+1|k} \delta\hat{X}_{k+1|k}^T \right\} \quad (13.75)$$

Insert Equation 13.70 into Equation 13.75 to get the Kalman covariance propagation equation

$$P_{k+1|k} = \bar{\Phi}_{k+1|k} P_{k|k} \bar{\Phi}_{k+1|k}^T + \Gamma_{k+1,k} Q_{k+1} \Gamma_{k+1,k}^T \quad (13.76)$$

where $\bar{\Phi}_{k+1|k}$ and $\Gamma_{k+1,k}$ are calculated according to Equations 13.68 and 13.73, and

$$Q_{k+1} = E \{ U_{k+1} U_{k+1}^T \} = \begin{bmatrix} \sigma^2 / (2a) & 0 \\ 0 & 0 \end{bmatrix} \quad (13.77)$$

because $E \{ Z_{k+1}^2 \} = 1$.

13.3 Other Stochastic Sequences

13.3.1 Brownian Motion

Random Walk Sequence

Random walk in one-dimension can be conveniently defined, and simulated, using a Wiener-Levy sequence (see Papoulis[93]). Let $\tau_0 > 0$ denote a fixed sample time, let $r_0 > 0$ denote a fixed constant, and let \mathbf{r}_i , $i \in \{1, 2, 3, \dots, n\}$, denote a random sequence defined by:

$$P\{\mathbf{r}_i = r_0\} = P\{\mathbf{r}_i = -r_0\} = 0.5 \quad (13.78)$$

where $P\{\mathbf{r}_i = r_0\} = 0.5$ is read: The probability that $\mathbf{r}_i = r_0$ is equal to 0.5.

Intuitively, the random variable \mathbf{r}_i could be described as the i^{th} toss of a fair coin: If heads, set $\mathbf{r}_i = r_0$; if tails set $\mathbf{r}_i = -r_0$, where r_0 is a fixed constant. By *fair*, we mean that the probability for heads is equal to 0.5, and the probability for tails is equal to 0.5. If we toss the coin a great number of times n , we expect: *mean* $\simeq \frac{1}{n} \sum_{i=1}^n \mathbf{r}_i = 0$, and *variance* $\simeq \frac{1}{n} \sum_{i=1}^n \mathbf{r}_i^2 = r_0^2$.

To simulate sequence \mathbf{r}_i , draw a uniformly distributed pseudo random number x_i repeatedly from the open interval $(0, 1)$. When $0 < x_i < 0.5$, set $\mathbf{r}_i = r_0$. When $0.5 \leq x_i < 1$, set $\mathbf{r}_i = -r_0$. From Eq. 13.78:

$$E\{\mathbf{r}_i\} = 0 \quad (13.79)$$

$$E\{\mathbf{r}_i^2\} = r_0^2 \quad (13.80)$$

Define a new random sequence $\bar{\mathbf{r}}(t)$ at n trials with:

$$\bar{\mathbf{r}}(t) = \sum_{i=1}^n \mathbf{r}_i = \sum_{i=1}^{t/\tau_0} \mathbf{r}_i \quad (13.81)$$

where:

$$t = n\tau_0 \quad (13.82)$$

Take the expectation of Eq. 13.81 and its square, and use Eqs. 13.79 and 13.80 to get:

$$E\{\bar{\mathbf{r}}(t)\} = 0 \quad (13.83)$$

$$E\{\bar{\mathbf{r}}^2(t)\} = nr_0^2 \quad (13.84)$$

because the \mathbf{r}_i are independent. Insert Eq. 13.82 into Eq. 13.84 to show:

$$\sigma_{\bar{\mathbf{r}}}^2(t) \equiv E\{\bar{\mathbf{r}}^2(t)\} = \left(\frac{r_0^2}{\tau_0}\right)t. \quad (13.85)$$

The random function $\bar{\mathbf{r}}(t) = \bar{\mathbf{r}}(n\tau_0)$ is called a *random walk* sequence, is easily simulated from the equations given, and is useful as it stands. However, there are two annoying properties: $\sigma_{\bar{\mathbf{r}}}^2(t)$ depends on the particular values selected for r_0 and τ_0 , and $\bar{\mathbf{r}}(t) = \bar{\mathbf{r}}(n\tau_0)$ is discontinuous at each step $t = n\tau_0$.

Wiener-Levy Process

We wish to take limits sending $r_0 \rightarrow 0$ and $\tau_0 \rightarrow 0$ simultaneously, so that $\sigma_{\bar{\mathbf{r}}}^2(t)$ will become independent of r_0 and τ_0 , and $\bar{\mathbf{r}}(t) = \bar{\mathbf{r}}(n\tau_0)$ will become continuous with time t . But r_0^2 goes to zero faster than τ_0 so that:

$$\lim_{\substack{r_0 \rightarrow 0 \\ \tau_0 \rightarrow 0 \\ t \text{ fixed}}} \left(\frac{r_0^2}{\tau_0} \right) t = 0 \quad (13.86)$$

a result that would contradict Eq. 13.85. This problem is solved by introducing constant $\alpha > 0$ and setting:

$$r_0 = \sqrt{\alpha\tau_0} \quad (13.87)$$

to control the relative rate at which $r_0 \rightarrow 0$. Define the Wiener-Levy process:

$$\mathbf{r}(t) = \lim_{\tau_0 \rightarrow 0} \bar{\mathbf{r}}(t) \quad (13.88)$$

Then:

$$\lim_{\tau_0 \rightarrow 0} \frac{r_0^2}{\tau_0} = \lim_{\tau_0 \rightarrow 0} \frac{\alpha\tau_0}{\tau_0} = \alpha \quad (13.89)$$

and the variance $\sigma_{\mathbf{r}}^2(t)$ on $\mathbf{r}(t)$:

$$\sigma_{\mathbf{r}}^2(t) \equiv E \{ \mathbf{r}^2(t) \} = \alpha t \quad (13.90)$$

For Proofs of the following results see Papoulis [93], Chapter 9. The Wiener-Levy process $\mathbf{r}(t)$ has zero mean:

$$E \{ \mathbf{r}(t) \} = 0 \quad (13.91)$$

and has a Gaussian density function:

$$f_{\mathbf{r}}(r; t) = \frac{1}{\sqrt{2\pi\alpha t}} \exp(-r^2/(2\alpha t)) \quad (13.92)$$

The covariance $R_{\mathbf{r}}(t_j, t_k)$ on $\mathbf{r}(t)$ is given by:

$$R_{\mathbf{r}}(t_j, t_k) \equiv E \{ \mathbf{r}(t_j) \mathbf{r}(t_k) \} = \alpha \cdot \min(t_j, t_k) \quad (13.93)$$

The Wiener-Levy process $\mathbf{r}(t)$ is continuous with time (except for negligible cases), and yet has independent increments:

$[\mathbf{r}(t_{k+2}) - \mathbf{r}(t_{k+1})]$ is independent of $[\mathbf{r}(t_{k+1}) - \mathbf{r}(t_k)]$ for all $k \in \{0, 1, 2, \dots\}$ (13.94)

The Wiener-Levy process $\mathbf{r}(t)$ is also called Brownian motion.

13.3.2 White Noise

Definition

Brownian motion $\mathbf{r}(t)$ has independent increments according to 13.94. White noise $\mathbf{w}_{k+1,k}$ is defined here in terms of these independent increments:

$$\mathbf{w}_{k+1,k} = \frac{\mathbf{r}(t_{k+1}) - \mathbf{r}(t_k)}{t_{k+1} - t_k}, \quad k \in \{0, 1, 2, \dots\} \quad (13.95)$$

$$t_k < t_{k+1} \quad (13.96)$$

Taking the Limit

According to Doob [19]:

$$\lim_{t_{k+1} \rightarrow t_k} \sup \{\mathbf{w}_{k+1,k}\} = \infty \quad (13.97)$$

where $\sup \{\mathbf{w}_{k+1,k}\}$ denotes the least upper bound of $\mathbf{w}_{k+1,k}$ over the set of all realizations. The associated variance is also infinite. Thus Brownian motion $\mathbf{r}(t)$ is not differentiable. The limit $t_{k+1} \rightarrow t_k$ on $\mathbf{w}_{k+1,k}$ is not realizable and is thereby useless for any physical representation. The difference $|t_{k+1} - t_k|$ may be small, but not arbitrarily small.

It is useful to attempt to simulate the limiting process. One easily finds that as $t_{k+1} \rightarrow t_k$, the magnitude of $\mathbf{w}_{k+1,k}$ gets large without bound. Simulation provides an easy method to understand Doob's theorem.

Mean

Run the expectation operator through Eq. 13.95 and use Eq. 13.91 to get:

$$E \{\mathbf{w}_{k+1,k}\} = 0, \quad k \in \{0, 1, 2, \dots\} \quad (13.98)$$

Variance

Define the white noise variance:

$$\sigma_{\mathbf{w}_{k+1,k}}^2 \equiv E \{\mathbf{w}_{k+1,k}^2\} \quad (13.99)$$

Insert Eq. 13.95 into Eq. 13.99:

$$\sigma_{\mathbf{w}_{k+1,k}}^2 = \frac{1}{(t_{k+1} - t_k)^2} [E \{ \mathbf{r}_k^2 \} - 2E \{ \mathbf{r}_k \mathbf{r}_{k+1} \} + E \{ \mathbf{r}_{k+1}^2 \}] \quad (13.100)$$

and use Eqs. 13.90 and 13.93:

$$\sigma_{\mathbf{w}_{k+1,k}}^2 = \frac{1}{(t_{k+1} - t_k)^2} [\alpha t_k - 2\alpha t_k + \alpha t_{k+1}] \quad (13.101)$$

to get:

$$\sigma_{\mathbf{w}_{k+1,k}}^2 = \frac{\alpha}{t_{k+1} - t_k} \quad (13.102)$$

Compare Eq. 13.102 to Eq. 13.90 to see the rigorous relation between white noise variance and Brownian motion variance with the common positive parameter α .

Covariance

Let us denote the white noise covariance $R_{\mathbf{w}} \equiv R_{\mathbf{w}}(t_j, t_{j+1}, t_k, t_{k+1})$, and define it with:

$$R_{\mathbf{w}} = E \{ \mathbf{w}_{j+1,j} \mathbf{w}_{k+1,k} \} \quad (13.103)$$

Then:

$$R_{\mathbf{w}} = \frac{\alpha}{t_{k+1} - t_k} \delta_{jk} \delta_{(j+1)(k+1)} \quad (13.104)$$

where δ_{mn} is the Kronecker delta. When $t_j = t_k$ and $t_{j+1} = t_{k+1}$:

$$R_{\mathbf{w}} = \sigma_{\mathbf{w}_{k+1,k}}^2 = \frac{\alpha}{t_{k+1} - t_k}$$

Otherwise:

$$R_{\mathbf{w}} = 0.$$

Physical Sources

From a *physical* perspective, white noise is derived from one known source: Resistivity in an electronic circuit. Thus electronic clocks, transmitters, receivers, and transponders accompany their signals with white noise. This physical process is called *thermal noise*, and is Gaussian as well as white. See Davenport and Root[99], page 185.

Filter Validation

One of the rigorous necessary conditions that derives from optimality of a sequential filter is that its measurement residuals are zero mean and white (Anderson and Moore [85], page 102). This is the basis for an important filter validation test, and is easily checked.

13.3.3 Clocks

Brownian Motion in Clock Phase

Recall Eq. 13.90 for the variance $\sigma_{\mathbf{r}}^2(t)$ on Brownian motion $\mathbf{r}(t)$:

$$\sigma_{\mathbf{r}}^2(t) = \alpha t$$

Associate a clock phase component with Brownian motion $\mathbf{r}(t)$, and use *seconds* for clock phase units. Then the variance $\sigma_{\mathbf{r}}^2(t)$ on Brownian motion has units seconds², t has units seconds, and α has units seconds.

White Noise in Clock Frequency

The frequency of this clock possesses a white noise component $\mathbf{w}_{k+1,k} = (\mathbf{r}(t_{k+1}) - \mathbf{r}(t_k)) / (t_{k+1} - t_k)$ associated with its Brownian motion $\mathbf{r}(t)$ in phase. Recall Eq. 13.102:

$$\sigma_{\mathbf{w}_{k+1,k}}^2 = \frac{\alpha}{t_{k+1} - t_k}$$

Thus the variance $\sigma_{\mathbf{w}_{k+1,k}}^2$ on clock frequency is unitless, as expected.

Allan Variance

Clock vendors provide an Allan Variance [1] diagram associated with each clock that quantifies white noise in fractional frequency.

Part III

Accelerations

Chapter 14

Accelerations

Recall Eqs. 1.21, 1.22, 1.23, 1.24, 1.25, and 1.26 for definitions relating to the discussion in this chapter. Then $\ddot{\mathbf{r}}$ denotes the total acceleration vector.

14.1 Two Body Acceleration

The equation for two-body acceleration $\ddot{\mathbf{r}}_K$ is well known:

$$\ddot{\mathbf{r}}_K = (-\mu/r^3) \mathbf{r} = [\mathbf{i}]^T (-\mu/r^3) z \quad (14.1)$$

The 3×1 matrix $(-\mu/r^3) z$ defines the two-body inertial acceleration matrix of cartesian components. Perturbative accelerations are defined as non-two-body accelerations, and are denoted herein with subscripts.

14.2 Total Acceleration

The total acceleration $\ddot{\mathbf{r}}$ has the additive decomposition:

$$\ddot{\mathbf{r}} = \ddot{\mathbf{r}}_K + \ddot{\mathbf{r}}_G + \ddot{\mathbf{r}}_{LS} + \ddot{\mathbf{r}}_{DL} + \ddot{\mathbf{r}}_P = [\mathbf{i}]^T [(-\mu/r^3) z + \ddot{z}_G + \ddot{z}_{LS} + \ddot{z}_{DL} + \ddot{z}_P] \quad (14.2)$$

where:

$\ddot{\mathbf{r}}_K = [\mathbf{i}]^T (-\mu/r^3) z$	two body
$\ddot{\mathbf{r}}_G = [\mathbf{i}]^T \ddot{z}_G$	Earth gravity
$\ddot{\mathbf{r}}_{LS} = [\mathbf{i}]^T \ddot{z}_{LS}$	luni-solar gravity
$\ddot{\mathbf{r}}_{DL} = [\mathbf{i}]^T \ddot{z}_{DL}$	air drag-lift
$\ddot{\mathbf{r}}_P = [\mathbf{i}]^T \ddot{z}_P$	solar photon pressure

Table 14.1: Accelerations

Chapter 15

Earth Gravity

15.1 Geopotential

Let U denote the geopotential function, defined by:

$$U = k^2 m_2 \int_{Earth} \frac{dm}{R}$$

where R is the length of the position vector \mathbf{R} with vector origin on spacecraft mass point m_2 and vector head on Earth mass differential dm , and where k^2 is the Gaussian gravitational constant. Laplace's partial differential equation is satisfied in the inertial component matrix X , and in the Earth-fixed component matrix Y :

$$\sum_{i=1}^3 \frac{\partial^2 U}{\partial X_i^2} = \sum_{i=1}^3 \frac{\partial^2 U}{\partial Y_i^2} = 0$$

where:

$$\mathbf{R} = [\mathbf{i}]^T X = [\mathbf{e}]^T Y$$

The transformation of Laplace's equation to orthogonal spherical coordinates was performed elegantly by Lass[60]:

$$\frac{\partial}{\partial r} \left[r^2 \frac{\partial U}{\partial r} \right] + \frac{1}{\cos \varphi} \frac{\partial}{\partial \varphi} \left[\cos \varphi \frac{\partial U}{\partial \varphi} \right] + \frac{1}{\cos^2 \varphi} \frac{\partial^2 U}{\partial \lambda^2} = 0$$

where r is the length of the position vector \mathbf{r} with vector origin on Earth center of mass and vector head on spacecraft mass point m_2 , where φ is the geocentric latitude, and λ is the Earth-fixed longitude. A solution to Laplace's equation in spherical coordinates was given by Kaula[51]:

$$U = \frac{\mu}{r} \sum_{n=0}^{\infty} \left(\frac{a_e}{r}\right)^n \sum_{m=0}^n P_{nm}(\sin \varphi) [C_{nm} \cos m\lambda + S_{nm} \sin m\lambda] \quad (15.1)$$

where $P_{n0}(\sin \varphi) = P_n(\sin \varphi)$ are Legendre polynomials, $P_{nm}(\sin \varphi)$ are associated Legendre functions, a_e is the equatorial radius of the reference Earth oblate ellipsoid, μ is the two-body gravitational constant with units distance-cubed per time-squared, and C_{nm} and S_{nm} are constants (integrals over mass dm) of degree n and order m . In practice, $n \leq N$ is truncated. Examples: LEO and HEO: $N = 50$. GEO: $N = 6$.

15.2 Legendre Functions

The purpose of this section is to aid in the development of geopotential acceleration error covariance functions. Let φ denote geocentric latitude, and let θ denote geocentric colatitude. In the definitions that follow, $x = \sin \varphi = \cos \theta$.

See Jahnke and Emde[47] for an excellent presentation of Legendre functions.

15.2.1 Legendre Polynomials of First Kind

Definition

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n, \quad x = \cos \theta, \quad n \in \{0, 1, 2, \dots\}$$

$$P_0(\cos \theta) = 1$$

$$P_1(\cos \theta) = \cos \theta$$

$$P_2(\cos \theta) = (3 \cos 2\theta + 1) / 4$$

$$P_3(\cos \theta) = (5 \cos 3\theta + 3 \cos \theta) / 8$$

$$P_4(\cos \theta) = (35 \cos 4\theta + 20 \cos 2\theta + 9) / 64$$

$$P_5(\cos \theta) = (63 \cos 5\theta + 35 \cos 3\theta + 30 \cos \theta) / 128$$

Table 15.1: Legendre Polynomials, Low Degree

Recursion

$$nP_n(x) = (2n - 1)xP_{n-1}(x) - (n - 1)P_{n-2}(x), \quad n \in \{2, 3, \dots\}$$

Theorem 3 All Legendre Polynomials $P_n(\cos \theta)$ are even functions of θ

15.2.2 Associated Legendre Functions of First Kind

Definition

$$P_{nm}(x) = (1-x^2)^{m/2} \frac{d^m P_n(x)}{dx^m}, \quad x = \cos \theta$$

$$m \in \{0, 1, 2, \dots, n\}, \quad n \in \{0, 1, 2, \dots\}$$

$$P_{22}(\cos \theta) = 3(1 - \cos 2\theta)/2$$

$$P_{32}(\cos \theta) = 15(\cos \theta - \cos 3\theta)/4$$

$$P_{42}(\cos \theta) = 15(3 + 4\cos 2\theta - 7\cos 4\theta)/16$$

Table 15.2: Associated Legendre Functions, Degree 2,3,4, Order 2

Recursion

$$(n-m)P_{nm}(x) = (2n-1)xP_{(n-1)m}(x) - (n+m-1)P_{(n-2)m}(x)$$

$$m \in \{0, 1, 2, \dots, n\}, \quad n \in \{0, 1, 2, \dots\}$$

Theorem 4 All Associated Legendre Functions $P_{n2}(\cos \theta)$ of order 2 are even functions of θ

15.3 Auto-Covariance Function on a Sphere

Let C_{nm} and S_{nm} denote geopotential coefficients of degree n and order m . Gravity acceleration errors of omission are incurred when any truncation of Eq. 15.1 is used. William Kaula defined[52] the degree n variance $\sigma_T^2(n)$, for each degree of truncation of Eq. 15.1, for his work in geodesy:

$$\sigma_T^2(n) = \sum_{m=0}^n (C_{nm}^2 + S_{nm}^2) \quad (15.2)$$

A similar function was defined by Wright[119] to account for gravity acceleration errors of omission:

$$\sigma_C^2(n) = \sum_{m=0}^n \left(E \left\{ (\delta C_{nm})^2 \right\} + E \left\{ (\delta S_{nm})^2 \right\} \right) \quad (15.3)$$

where δC_{nm} and δS_{nm} are estimation errors for estimates of C_{nm} and S_{nm} . Combine these functions to define:

$$\sigma_n^2 = \begin{cases} \sigma_T^2(n), & n > N \\ \sigma_C^2(n), & n \leq N \end{cases} \quad (15.4)$$

Kaula also constructed gravity acceleration error auto-covariance functions on a sphere with constant radius r that encloses the Earth's surface. Let us denote a 3×1 gravity acceleration error matrix δg in RIC components with:

$$\delta g = \begin{bmatrix} \delta g_R \\ \delta g_I \\ \delta g_C \end{bmatrix} \quad (15.5)$$

Assume δg to be unbiased and Gaussian distributed. For a fixed point P_0 on the sphere, and associated 3×1 gravity acceleration error matrix δg_0 in RIC components, and for each point P on the sphere and its associated 3×1 gravity error matrix δg , Kaula derived three auto-covariance expressions for P_0 by averaging $E \{ \delta g(P_0) \delta g^T(P) \}$ over the sphere:

$$\sigma_{RR}^2(\psi) = \sum_n \left[\frac{n+1}{n-1} \right]^2 \left[\frac{a_e}{r} \right]^{2n+4} P_{n0}(\cos \psi) \sigma_n^2 \quad (15.6)$$

$$\sigma_{II}^2(\psi) = \frac{1}{2} \sum_n \left[\frac{n(n+1)}{(n-1)^2} \right] \left[\frac{a_e}{r} \right]^{2n+4} \left[P_{n0}(\cos \psi) - \frac{P_{n2}(\cos \psi)}{n(n+1)} \right] \sigma_n^2 \quad (15.7)$$

$$\sigma_{CC}^2(\psi) = \frac{1}{2} \sum_n \left[\frac{n(n+1)}{(n-1)^2} \right] \left[\frac{a_e}{r} \right]^{2n+4} \left[P_{(n-1)0}(\cos \psi) + \frac{P_{(n-1)2}(\cos \psi)}{n(n+1)} \right] \sigma_n^2 \quad (15.8)$$

for the sphere, where ψ is the central angle between P_0 and each P . This result was extended by Kay Pechenick[96] to include the symmetric off-diagonal covariance expressions:

$$\Gamma_{RI}(\psi) = -\frac{1}{2} \sum_n \left[\frac{n(n+1)^2}{(n-1)^2} \right] \left[\frac{a_e}{r} \right]^{2n+4} \left[P_{(n-1)0}(\cos \psi) + \frac{P_{(n-1)2}(\cos \psi)}{n(n+1)} \right] (\sin \psi) \sigma_n^2 \quad (15.9)$$

$$\Gamma_{IR}(\psi) = \Gamma_{RI}(\psi) \quad (15.10)$$

$$\Gamma_{CR}(\psi) = \Gamma_{RC}(\psi) = 0 \quad (15.11)$$

$$\Gamma_{CI}(\psi) = \Gamma_{IC}(\psi) = 0 \quad (15.12)$$

The complete auto-covariance 3×3 matrix function has the representation:

$$R(\psi) = \begin{bmatrix} \sigma_{RR}^2(\psi) & \Gamma_{RI}(\psi) & \Gamma_{RC}(\psi) \\ \Gamma_{IR}(\psi) & \sigma_{II}^2(\psi) & \Gamma_{IC}(\psi) \\ \Gamma_{CR}(\psi) & \Gamma_{CI}(\psi) & \sigma_{CC}^2(\psi) \end{bmatrix} \quad (15.13)$$

Theorem 5 $R_{11}(\psi) = \sigma_{RR}^2(\psi)$, $R_{22}(\psi) = \sigma_{II}^2(\psi)$, and $R_{33}(\psi) = \sigma_{CC}^2(\psi)$ are even functions of ψ , because they are linear combinations of Legendre Polynomials and Associated Legendre Functions of order 2. That is: $R_{jj}(-\psi) = R_{jj}(\psi)$, $j \in \{1, 2, 3\}$. This is a necessary condition for any stationary auto-covariance function $R_{jj}(\psi)$.

15.3.1 From Angle to Time

Let n denote mean orbit motion, a two-body constant. Let t and τ denote arbitrary times, and set:

$$[t - \tau]n = \psi \quad (15.14)$$

Then:

$$R(\psi) = R([t - \tau]n) \quad (15.15)$$

Notice that:

$$R([t - \tau]n) = R([(t + T) - (\tau + T)]n) \quad (15.16)$$

Auto-covariance $R([t - \tau]n)$ may be translated by any time increment T . It is therefore called *stationary*.

15.3.2 Covariance Function $R(0)$

The components of $R(0)$ are required:

$$\sigma_{RR}^2(0) = \sum_n \left[\frac{n+1}{n-1} \right]^2 \left[\frac{a_e}{r} \right]^{2n+4} \sigma_n^2 \quad (15.17)$$

$$\sigma_{II}^2(0) = \frac{1}{2} \sum_n \left[\frac{n(n+1)}{(n-1)^2} \right] \left[\frac{a_e}{r} \right]^{2n+4} \sigma_n^2 \quad (15.18)$$

$$\sigma_{CC}^2(0) = \frac{1}{2} \sum_n \left[\frac{n(n+1)}{(n-1)^2} \right] \left[\frac{a_e}{r} \right]^{2n+4} \sigma_n^2 \quad (15.19)$$

$$\Gamma_{RI}(0) = 0 \quad (15.20)$$

15.3.3 Auto-correlation Function

When $\psi = 0$, the Legendre polynomials are unity, the associated Legendre functions are zero, the instantaneous matrix $R(0)$ is diagonal and has non-zero diagonal elements. This enables the definition of a diagonal 3×3 matrix auto-correlation function:

$$\rho(\psi) = [R(0)]^{-1/2} R(\psi) [R(0)]^{-1/2} \quad (15.21)$$

and it enables the multiplicative decomposition of $R(\psi)$:

$$R(\psi) = [R(0)]^{1/2} \rho(\psi) [R(0)]^{1/2} \quad (15.22)$$

It is important here to notice that the effect of r on $\rho(\psi)$ is divided out, so that significant dynamics due to variations in r are represented in $R(\psi)$ by $[R(0)]^{1/2}$. This is important for implementation.

15.4 Acceleration Errors on a Sphere

For each fixed point P_0 on a sphere of radius $r > a_e$, Eqs. 15.6 through 15.13 provide a geopotential covariance function $R(\psi)$, derived by averaging the product of acceleration error δg_0 at P_0 with every other acceleration error δg (point P) on the sphere. Consider now the intersection of this sphere with a plane that contains the origin of the sphere. This intersection is a circle with fixed radius r . Refer to this circle as a reference circular spacecraft orbit with semi-major axis $a = r$, eccentricity $e = 0$, spacecraft speed $\dot{s} = \sqrt{\mu/a}$, and two plane orientation parameters; *e.g.*, inclination i and node Ω . Consider a sequence of P_0 points, call them P_k , $k \in \{0, 1, 2, \dots\}$, where each point lies on the circular orbit, is associated with a time t_k , and has a covariance matrix function $R^k(\psi)$ at t_k . Our task in this section is to integrate the 3×3 gravity acceleration error covariance matrices $R^k(\psi)$ with time so as to form 6×6 orbit error covariance matrices $P_{1,k+1,k}^{\int}$, and to accumulate these into a running sequential sum of orbit error covariance matrices $P_{k+1|k}$.

The linear integral error model, particularized to the gravity acceleration error 3×1 matrix δg and the orbit error 6×1 matrix δZ , is given by:

$$\delta Z(t_{k+1}) = \Phi(t_{k+1}, t_k) \delta Z(t_k) + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) G(\tau) \delta g(\tau) d\tau \quad (15.23)$$

$$k \in \{0, 1, 2, \dots\}$$

where matrix $\Phi(t_{k+1}, \tau)$ is a 6×6 orbit transition matrix, and matrix $G(\tau)$ is a 6×3 VOP (Variation Of Parameters) matrix of partial derivatives that transforms acceleration variations to orbit variations. If $t_{k+1} = t_k$, then there is no contribution to $\delta Z(t_{k+1})$ from any $\delta g(\tau)$; *i.e.*, it takes time (integration with time) to move acceleration errors to orbit errors. Each gravity acceleration error $\delta g(\tau)$ is random, therefore each orbit error $\delta Z(t_{k+1})$ is also random. The orbit error covariance filter time-update matrix, using the expectation operator $E\{\cdot\}$, is defined by:

$$P_{k+1|k} = E\left\{\delta Z(t_{k+1}) \delta Z(t_{k+1})^T\right\} \quad (15.24)$$

Then:

$$P_{k+1|k} = \Phi(t_{k+1}, t_k) P_{k|k} \Phi(t_{k+1}, t_k)^T + P_{k+1,k}^{\int} \quad (15.25)$$

where:

$$P_{k+1,k}^{\int\int} = \sum_j P_{j,k+1,k}^{\int\int} \quad (15.26)$$

$$P_{1,k+1,k}^{\int\int} = I_{k+1,k}^C + I_{k+1,k}^L + I_{k+1,k}^R \quad (15.27)$$

$$I_{k+1,k}^C = \int \int_{t_k}^{t_{k+1}} H(t_{k+1}, \tau) E \{ \delta g(\tau) \delta g^T(t) \} H^T(t_{k+1}, t) d\tau dt \quad (15.28)$$

$$I_{k+1,k}^L = \Phi(t_{k+1}, t_k) \int_{t_k}^{t_{k+1}} E \{ \delta Z(t_k) \delta g^T(t) \} H^T(t_{k+1}, t) dt \quad (15.29)$$

$$I_{k+1,k}^R = \int_{t_k}^{t_{k+1}} H(t_{k+1}, \tau) E \{ \delta g(\tau) \delta Z^T(t_k) \} d\tau \Phi^T(t_{k+1}, t_k) \quad (15.30)$$

$$H(t_{k+1}, \eta) = \Phi(t_{k+1}, \eta) G(\eta) \quad (15.31)$$

If a measurement is processed at time t_k by the sequential filter, then Eq. 15.23 should be written:

$$\delta Z(t_{k+1}|t_k) = \Phi(t_{k+1}, t_k) \delta Z(t_k|t_k) + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) G(\tau) \delta g(\tau) d\tau \quad (15.32)$$

$$k \in \{0, 1, 2, \dots\}$$

to explicitly indicate the use of new information at time t_k . This reduces the Markov problem by partially decoupling $\delta Z(t_k|t_k)$ and $\delta g(\tau)$, because $\tau \geq t_k$.

15.4.1 Computational Problem

Computer experiments were performed in 1979 at the General Electric Company in King of Prussia to evaluate $I_{k+1,k}^C$ numerically in double precision for $t_{k+1} - t_k = 5$ min on a sphere with iterated integrals, using Eq. 15.15 for evaluation of $R(\psi) = R([t - \tau]n) = E \{ \delta g(\tau) \delta g^T(t) \}$. This 5 minute accumulation took 3 days to complete on an IBM 360, unacceptable for operational use. The numerically evaluated 6×6 matrix result was approximately symmetric in the upper left-hand corner, but was very significantly unsymmetric in the lower right-hand corner.

15.4.2 Solution to Computational Problem

The computational problem just identified led to development of the following approximation[119] of Eq. 15.27:

$$P_{1,k+1,k}^{\int \int} \approx II = \int_{t_k}^{t_{k+1}} J(t_{k+1}, \tau) \bar{I}_G J^T(t_{k+1}, \tau) d\tau \quad (15.33)$$

$$k \in \{0, 1, 2, \dots\}$$

where:

$$J(t_{k+1}, \tau) = \Phi(t_{k+1}, \tau) G(\tau) [R(0)]^{1/2} \quad (15.34)$$

$$I_G(\psi) = \frac{2}{n} \left\{ GRAPH_{(0 \leq \psi \leq \pi)} \int_0^\psi \rho(\beta) d\beta \right\} \quad (15.35)$$

$$\bar{I}_G: \text{select an appropriate } 3 \times 3 \text{ matrix value from graph } I_G(\psi) \quad (15.36)$$

15.4.3 The Double Integral

The left-hand side of Eq. 15.33 is a double integral according to Eqs. 15.26 through 15.31. The right-hand side of Eq. 15.33 is also a double integral as seen in Eq. 15.35, where:

$$2 \int_0^\psi \rho(\beta) d\beta = \int_{-\psi}^\psi \rho(\beta) d\beta$$

The latter equality derives from the fact that the diagonal elements of $\rho(\psi)$ are symmetric about the origin $\psi = 0$. The diagonal elements of $\rho(\psi)$ also are maximum and positive at the origin, required by definition of any stationary auto-correlation function. The inner integral \bar{I}_G , defined by Eq. 15.36, is an analytic time constant, enabled by the property that the effect of $r(t)$ on $\rho(\psi)$ has been divided out, so that sensitivity of $R(\psi)$ to $r(t)$ is captured by $[R(0)]$ in the outer integral of Eq. 15.33. The calculation of II generates a symmetric and positive definite orbit error covariance matrix when the three diagonal values of \bar{I}_G are positive.

15.4.4 Markov Property

Optimal Orbit Determination

Given the state estimate and error covariance at time t_k , we require computation of the state estimate and its error covariance at any later time t_{k+1} without referring to information at times prior to t_k . This is the essence of the Markov property. It enables computational tractability for sequential filtering. More important, it enables *optimal* orbit determination when tracking measurements are acquired sequentially with time.

Application of Feller's Definition to Orbit Errors

Recall Eq. 15.23. To paraphrase Feller[21]: $\delta Z(t_j)$, $j, k \in \{0, 1, 2, \dots\}$, is a Markov sequence if, given the orbit error $\delta Z(t_k)$ at time t_k , no additional data concerning orbit errors at previous times (say $\delta g(\tau)$, for $\tau < t_k$) can alter the conditional probability for the orbit error $\delta Z(t_{k+1})$ at any future time t_{k+1} . Then Eq. 15.23 is a Gauss-Markov sequence.

Orbit Error Problem

Eq. 15.23, for $k \rightarrow k - 1$, can be written:

$$\delta Z(t_k) = \Phi(t_k, t_{k-1}) \delta Z(t_{k-1}) + \int_{t_{k-1}}^{t_k} \Phi(t_k, \tau) G(\tau) \delta g(\tau) d\tau \quad (15.37)$$

The covariance $I_{k+1,k}^L$ of Eq. 15.29 is required formally to estimate the propagation $\delta Z(t_{k+1})$. Notice that $\delta Z(t_k)$ is required in $I_{k+1,k}^L$ of Eq. 15.29, so that gravity acceleration errors $\delta g(\tau)$ for $t_{k-1} < \tau < t_k$ are required to evaluate $\delta Z(t_{k+1})$, given $\delta Z(t_k)$ at time t_k . And so $\delta Z(t_k)$ is non-Markov because $\delta g(\tau)$, $[t_{k-1} < \tau < t_k]$, is required from times previous to t_k , and this does alter the conditional probability for the orbit error $\delta Z(t_{k+1})$ at future time t_{k+1} .

Solution to Orbit Error Problem

The inner integral $I_G(\psi)$ of Eq. 15.35 is defined with respect to Earth-fixed central angle ψ , and is thus independent of any time. The outer integral II depends on data from the time interval $[t_k, t_{k+1}]$, and not on any data prior to time t_k . The covariance approximation given by Eq. 15.33 is therefore Markov. It is also Gaussian because it approximates the second moment to the Gaussian sequence Eq. 15.23.

15.4.5 Inertial Frame

William Kaula has given the gravity acceleration error covariance in the Gaussian RIC trajectory frame, and this frame rotates at orbit angular speed, mean motion n . It is non-inertial. Eq. 15.33 presents an orbit error covariance, integrated to generate position and velocity components in the ICRF inertial frame. How do we get from the Gaussian non-inertial frame to the ICRF inertial frame?

The answer is given by the integrand $\Phi(t_{k+1}, \tau) G(\tau) \delta g(\tau)$ of Eq. 15.23. The acceleration error 3×1 gravity error matrix $\delta g(\tau)$ is defined on the Gaussian non-inertial frame. The 6×3 matrix $G(\tau)$ transforms $\delta g(\tau)$ to a 6×1 matrix $G(\tau) \delta g(\tau)$ in the ICRF inertial frame. The 6×6 transition matrix $\Phi(t_{k+1}, \tau)$ integrates $G(\tau) \delta g(\tau)$ from time τ to time t_{k+1} . So the integration with time is performed in the ICRF inertial frame.

15.4.6 Stochastic Characterization

For random gravity modeling errors, we have demonstrated that both our orbit stochastic model Eq. 15.23 and our orbit covariance function implementation Eq. 15.33 are Gauss-Markov. But thermal white-noise sequences are Gauss-Markov, random walk sequences are Gauss-Markov, summed (generalized integral) random walk sequences are Gauss-Markov, and we have used exponentially correlated Gauss-Markov sequences. How can we characterize our orbit Gauss-Markov sequence for random gravity modeling errors?

Our orbit Gauss-Markov sequence for random gravity modeling errors is similar to a six-dimensional cross-correlated random walk, but it is shaped and scaled by the physics of the geopotential function.

15.4.7 Implementation

Calculations A Priori

- A value of orbit period P is selected to represent the LEO class of interest at perigee.
- Calculate $n = 2\pi/P$ and $r = a = (\mu/n^2)^{1/3}$, with constraint $e = 0$
- Calculate $I_G(\psi)$ using Eqs. 15.35 and 15.21
- Generate three graphs for the diagonal scalar functions of the 3×3 matrix function $I_G(\psi)$
- Derive a 3×3 diagonal matrix of numbers \bar{I}_G (units minutes), by appropriate selection from $I_G(\psi)$

Calculations During Each Sequential Filter Run

Given the time interval $[t_k, t_{k+1}]$ for filter time-update, and given current osculating values for r :

- Calculate $P_{1,k+1,k}^{\int \int}$ according to Eq. 15.33 using \bar{I}_G to approximate $I_G(\psi)$, and using the current r for $R(0)$
- Add the gravity term $P_{1,k+1,k}^{\int \int}$ to those for solar pressure, atmospheric density, and thrusting to form the sum $P_{k+1,k}^{\int \int}$
- Calculate $P_{k+1|k}$ according to Eq. 15.25 for each filter time-update

15.5 Perigee-Apogee Weighting

Define the class of Low Earth Orbits (LEO) as those with radial distances r such that $1.04 \text{ er} < r \leq 1.2 \text{ er}$. The lower bound relates to keeping a spacecraft in free-fall without thrusting for a few orbits, and the upper bound is associated

with TOPEX/JASON high-LEOs. The orbit eccentricity is then bounded: $0 < e \leq 0.071\dots$. This captures the majority of *physical* low Earth orbits. LEOs are then necessarily near-circular orbits.

Let us denote the orbit true anomaly with v . Since there are no *physical* LEOs with zero mean eccentricity¹, we shall not worry about existence of the associated true anomaly v in that which follows. Let $C = C(v)$ denote the gravity acceleration error covariance 3×3 matrix function at the true anomaly v . Unfortunately, our method to calculate $C(v)$ is rigorously valid only for circular LEO orbits², and for this case $C(v) = C$ is independent of v . So right away we have a disconnect between physical orbits and our method. Here we live in a world of approximations.

It is suspected that sources of geopotential-modeling error covariance representations are not yet able to provide representations that are *physically* accurate to better than two significant digits. Under this assumption, we are justified in making any approximation that preserves two decimals in covariance transformations derived therefrom. More important, it seems we only require two significant digits to construct an effective sequential filter orbit gain from the acceleration covariance.

Gravity acceleration modeling errors are sensitive to the radial component of orbit position. In an attempt to use a zero-eccentricity theory with physical non-zero eccentricity orbits, I have represented the LEO with increasing true anomaly as passing through an infinite number of circular orbits, each with a distinct radial component of position, during each orbit period³.

Let C_P denote the gravity acceleration error covariance 3×3 matrix function on a sphere with perigee radius, and let C_A denote the gravity acceleration error covariance 3×3 matrix function on a sphere with apogee radius. It is computationally convenient to calculate $C_P = C(0)$ and $C_A = C(\pi)$, and store them equivalently as 3×3 diagonal matrix constants to be used with weighting functions to approximate $C(v)$.

15.5.1 Weighting Functions

Let us denote the perigee weighting function with $P(v)$, denote the apogee weighting function with $A(v)$, and require:

$$P(v) + A(v) = 1 \quad (15.38)$$

Given the current value of true anomaly v , calculate the approximation $\hat{C}(v)$ to $C(v)$:

$$\hat{C}(v) = P(v)C_P + A(v)C_A \quad (15.39)$$

¹Use the sample mean eccentricity for any LEO, averaged over all osculating eccentricities for one orbit period.

²I have an approach for building a theory and model for HEOs, but have not taken the time to accomplish this.

³The application of this method to HEOs fails the McReynolds filter-smoother consistency test.

if the osculating orbit eccentricity e satisfies:

$$e > \epsilon_e$$

Otherwise, use:

$$C(v) = C_P$$

For perigee define $P(v)$:

$$P(v) = (1 + \cos v)/2 \quad (15.40)$$

For apogee define $A(v)$:

$$A(v) = (1 - \cos v)/2 \quad (15.41)$$

15.6 New Method

15.6.1 Splines

An option has been implemented and validated that enables the user to replace the perigee-apogee inner integral calculation of Section 15.5 with general polynomial splines⁴. The class of LEO near circular orbits approximated with the perigee-apogee inner integral is small compared to the class of LEO near circular orbits approximated with the new polynomial splines. The arbitrary potential function truncation by the user at $(\text{deg,ord}) = (N,N)$ invokes errors of truncation (omission) and errors of commission. The inner integral splines are designed to accommodate both types of errors.

15.6.2 Sample Covariance Validation

A new sample covariance technique was created to quantify comparisons of the filter gravity error process noise covariance function to a propagated sample covariance derived from numerical trajectory integration simulations⁵. Perturbative accelerations for the trajectory integration simulations are created as in Eq. 15.58 of Section 15.7.

15.6.3 GRACE

An option has been implemented to apply a filter gravity error process noise covariance function developed from the GRACE geopotential error covariance matrix. This option uses the new polynomial splines. Validation of the GRACE

⁴See the paper[130] entitled *Orbit Covariance Inner Integrals With Splines*.

⁵See the paper[129] entitled *Consistency of Sample Orbit Covariance Function with Acquired Potential Covariance Matrix*.

geopotential error covariance matrix, using McReynolds' filter-smoother consistency test without filter scaling, was successful⁶.

15.6.4 EGM96

An option has been implemented to apply a filter gravity error process noise covariance function developed from the EGM96 geopotential error covariance matrix. This option uses the new polynomial splines. Validation of the EGM96 geopotential error covariance matrix, using McReynolds' filter-smoother consistency test without filter scaling, failed⁷ for the crosstrack position component. An acceptable validation was performed by scaling the crosstrack component in the ODTK filter.

15.7 Simulate Gravity Coefficient Errors

15.7.1 Covariance Matrix Representations

Let \mathbf{x} denote an $n \times 1$ column matrix random variable of Gaussian numbers, let I denote the identity matrix, and let $E\{\cdot\}$ denote the linear expectation operator. Assume that:

$$E\{\mathbf{x}\} = 0 \quad (15.42)$$

Define the covariance $n \times n$ matrix P of x with:

$$P = E\{\mathbf{x}\mathbf{x}^T\} \quad (15.43)$$

Each covariance matrix P has a decomposition:

$$P = M\Lambda M^T \quad (15.44)$$

where the $n \times n$ modal matrix M is orthogonal:

$$MM^T = M^T M = I \quad (15.45)$$

where the $n \times 1$ column matrices of M are unit eigenvectors:

$$M = \begin{bmatrix} M_1 & M_2 & M_3 & \dots & M_n \end{bmatrix} \quad (15.46)$$

and where Λ is an $n \times n$ matrix of off-diagonal zeros and diagonal eigenvalues $\lambda_j \geq 0$, $j \in \{1, 2, 3, \dots, n\}$:

⁶See the paper[128] entitled *Orbit Gravity Error Covariance Functions - GRACE, EGM96, & Lunar Prospector*.

⁷See the paper[129] entitled *Consistency of Sample Orbit Covariance Function with Acquired Potential Covariance Matrix*, 18th AAS/AIAA Space Flight Mechanics Meeting, January 2008.

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ 0 & 0 & \lambda_3 & & \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & & & \lambda_n \end{bmatrix} \quad (15.47)$$

Each covariance matrix P has a spectral representation:

$$P = \sum_{j=1}^n \lambda_j M_j M_j^T \quad (15.48)$$

15.7.2 Simulation of \mathbf{x}

Theory

Let \mathbf{g} denote an $n \times 1$ column matrix random variable of white Gaussian elements with zero mean and unit variance. Then:

$$E\{\mathbf{g}\} = 0 \quad (15.49)$$

$$E\{\mathbf{g}\mathbf{g}^T\} = I \quad (15.50)$$

Define $\Lambda^{1/2}$ and random variable \mathbf{y} :

$$\Lambda^{1/2} = \begin{bmatrix} \lambda_1^{1/2} & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2^{1/2} & 0 & \cdots & 0 \\ 0 & 0 & \lambda_3^{1/2} & & \\ \vdots & \vdots & & \ddots & \\ 0 & 0 & & & \lambda_n^{1/2} \end{bmatrix} \quad (15.51)$$

$$\mathbf{y} = M\Lambda^{1/2}\mathbf{g} \quad (15.52)$$

Then:

$$E\{\mathbf{y}\} = M\Lambda^{1/2}E\{\mathbf{g}\} = 0 \quad (15.53)$$

$$E\{\mathbf{y}\mathbf{y}^T\} = M\Lambda^{1/2}E\{\mathbf{g}\mathbf{g}^T\}\Lambda^{1/2}M^T = M\Lambda M^T = P \quad (15.54)$$

There is no way to distinguish random variable \mathbf{y} from random variable \mathbf{x} . Therefore set:

$$\mathbf{x} = M\Lambda^{1/2}\mathbf{g} \quad (15.55)$$

Practice

Let g_k and x_k denote the k^{th} realizations of random variables \mathbf{g} and \mathbf{x} . Then from Eq. 15.55:

$$x_k = M\Lambda^{1/2}g_k, \quad k \in \{1, 2, 3, \dots, N\} \quad (15.56)$$

Use Eq. 15.56 to simulate x_k . Validate with Eq. 15.43:

$$P = E\{\mathbf{xx}^T\} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N x_k x_k^T \quad (15.57)$$

15.7.3 A Useful Example

Given a column matrix \hat{X} of estimated geopotential coefficients, and given an associated geopotential coefficient error covariance matrix P , calculate an ensemble of simulated column matrices X_k , $k \in \{1, 2, 3, \dots, N\}$, of geopotential coefficients.

Use Eq. 15.56 to simulate x_k , a column matrix of simulated errors of geopotential coefficients. Then:

$$X_k = \hat{X} + x_k, \quad k \in \{1, 2, 3, \dots, N\} \quad (15.58)$$

Chapter 16

Lunar Gravity

With the exception of Section 15.5, the technical modeling content of Chapter 15 on Earth Gravity is applicable to this chapter on Lunar Gravity, with corresponding references to the lunar gravity field in place of that of the Earth. Therefore this material will not be repeated.

16.1 Lunar Prospector

A new lunar orbit gravity process noise covariance function has been developed from the Lunar Prospector potential covariance matrix[55] to enable sequential filtering and smoothing of lunar tracking data[128]. This option uses the new polynomial spline technique referred to in Section 15.6.1. Validation of the lunar orbit gravity process noise covariance function for errors of commission was performed using the new sample covariance comparison technique[129]. The lunar orbit gravity process noise covariance function was shown to generate a stable filter for lunar sequential orbit determination.

Chapter 17

Atmospheric Drag and Lift

17.1 King-Hele Unit Vector \mathbf{K}

King-Hele[53] asserts, on average¹, that the atmosphere co-rotates with the Earth. Define the Earth angular velocity with:

$$\boldsymbol{\omega} = [\mathbf{i}]^T \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} \quad (17.1)$$

where ω is Earth angular speed. Let us denote spacecraft velocity with respect to the atmosphere with $\dot{\mathbf{r}}$. Then from mechanics:

$$\dot{\mathbf{r}} = \dot{\mathbf{r}} - \boldsymbol{\omega} \times \mathbf{r} \quad (17.2)$$

Define the velocity magnitude with:

$$\dot{s} = \sqrt{\dot{\mathbf{r}} \cdot \dot{\mathbf{r}}} \quad (17.3)$$

Now define the King-Hele unit vector:

$$\mathbf{K} = \frac{\dot{\mathbf{r}}}{\dot{s}} = [\mathbf{i}]^T K \quad (17.4)$$

with inertial components contained by:

$$K = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \end{bmatrix} \quad (17.5)$$

¹I asked Luigi Jacchia (in the mid 70s) how to define/construct an atmospheric density error model and covariance for LEO orbit determination. In his initial irritated response, Jacchia referred to random longitude dependent winds at LEO heights, blowing from the Earth's poles to its equator at speeds greater than Mach 1. But finally he referred to King-Hele for a place to begin: Assume the atmosphere co-rotates with the Earth.

17.2 Orthonormal Vector Basis $[\mathbf{a}]$

An appropriate orthonormal vector basis, call it $[\mathbf{a}]$, is required for representation of atmospheric drag and lift accelerations:

$$[\mathbf{a}] = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} \quad (17.6)$$

Unit vector \mathbf{K} defines the direction of the spacecraft velocity vector relative to the mean rotating atmosphere. Therefore it is appropriate to anchor $[\mathbf{a}]$ to \mathbf{K} :

$$\mathbf{a}_2 = \mathbf{K} = [\mathbf{i}]^T K = (a_{21}, a_{22}, a_{23}) [\mathbf{i}] \quad (17.7)$$

Use:

$$\mathbf{U} = \mathbf{r}/r$$

to construct:

$$\mathbf{\Lambda} = \mathbf{U} \times \mathbf{K} \quad (17.8)$$

Define:

$$\mathbf{a}_3 = \mathbf{\Lambda}/\lambda = (a_{31}, a_{32}, a_{33}) [\mathbf{i}] \quad (17.9)$$

where:

$$\lambda = \sqrt{\mathbf{\Lambda} \cdot \mathbf{\Lambda}} \quad (17.10)$$

Complete the orthonormal basis with:

$$\mathbf{a}_1 = \mathbf{a}_2 \times \mathbf{a}_3 = (a_{11}, a_{12}, a_{13}) [\mathbf{i}] \quad (17.11)$$

Then:

$$\begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} \mathbf{i}_1 \\ \mathbf{i}_2 \\ \mathbf{i}_3 \end{bmatrix} \quad (17.12)$$

That is:

$$[\mathbf{a}] = R_{ai} [\mathbf{i}] \quad (17.13)$$

where:

$$R_{ai} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (17.14)$$

17.3 Drag-Lift Acceleration

The air drag-lift acceleration $\ddot{\mathbf{r}}_{DL}$ is defined by:

$$\ddot{\mathbf{r}}_{DL} = [\mathbf{a}]^T \ddot{z}_{DL}^a \quad (17.15)$$

Insert Eq. 17.13 into Eq. 17.15 to get:

$$\ddot{\mathbf{r}}_{DL} = [\mathbf{i}]^T R_{ai}^T \ddot{z}_{DL}^a = [\mathbf{i}]^T R_{ia} \ddot{z}_{DL}^a \quad (17.16)$$

and compare this to:

$$\ddot{\mathbf{r}}_{DL} = [\mathbf{i}]^T \ddot{z}_{DL} \quad (17.17)$$

to write:

$$\ddot{z}_{DL} = R_{ia} \ddot{z}_{DL}^a \quad (17.18)$$

where drag-lift acceleration components are identified with:

$$\ddot{z}_{DL}^a = \begin{bmatrix} \ddot{z}_{LR}^a \\ \ddot{z}_{DI}^a \\ \ddot{z}_{LC}^a \end{bmatrix} = \begin{bmatrix} \text{lift component radial} \\ \text{drag component intrack} \\ \text{lift component crosstrack} \end{bmatrix} \quad (17.19)$$

Assume that integrals of the lift components \ddot{z}_{LR}^a and \ddot{z}_{LC}^a , and their errors, are negligible² relative to those of the scalar drag component \ddot{z}_{DI}^a , and replace notation for the 3×1 drag acceleration matrix \ddot{z}_{DL}^a with \ddot{z}_D^a . That is:

$$\ddot{z}_D^a = \begin{bmatrix} 0 \\ \ddot{z}_{DI}^a \\ 0 \end{bmatrix} \quad (17.20)$$

Let \ddot{z}_D denote the associated 3×1 drag acceleration matrix in inertial components. Then from Eq. 17.18:

$$\ddot{z}_D = R_{ia} \ddot{z}_D^a \quad (17.21)$$

17.4 Air-Drag Acceleration

17.4.1 Drag Acceleration

The perturbative air-drag acceleration matrix estimate has the form:

$$\ddot{z}_D = -\frac{1}{2} B \hat{\rho} \hat{s}^2 K \quad (17.22)$$

where:

²Integrals, and their errors, of the intrack component \ddot{z}_{DI}^a accumulate secularly into the orbit mean motion and mean longitude estimates rev after rev, whereas integrals and errors of the radial and crosstrack lift components \ddot{z}_{LR}^a and \ddot{z}_{LC}^a are quasi-periodic and have significant cancelling within each orbit period.

$$B = \frac{C_D A}{m} \quad (17.23)$$

where \ddot{z}_D is a 3×1 perturbative drag acceleration matrix with inertial components, where $\hat{\rho} = \hat{\rho}(h)$ is the atmospheric density estimate calculated according to Eq. 17.52, C_D is a unitless drag coefficient, A is the spacecraft area projection onto a plane orthogonal to the spacecraft velocity vector $\hat{\mathbf{r}}$ referred to a rotating Earth, \hat{s} is the length of $\hat{\mathbf{r}}$, m is spacecraft mass, and K is the King-Hele 3×1 unit matrix defined above that contains inertial components of $\hat{\mathbf{r}}/\hat{s}$.

17.5 Ballistic Coefficient Errors

The estimation errors $\Delta\ddot{z}_D$ in LEO air-drag accelerations \ddot{z}_D are due most significantly to random errors in modeled atmospheric density ρ and ballistic coefficient B . Differentiation of Eq. 17.22 provides:

$$\Delta\ddot{z}_D = \left(\frac{\Delta B}{B} + \frac{\Delta\rho}{\rho} \right) \ddot{z}_D \quad (17.24)$$

an associated instantaneous error model. We model both $\Delta B/B$ and $\Delta\rho/\rho$ as exponentially correlated Gauss-Markov sequences. When their exponential half-life times are significantly different, then they can be estimated simultaneously. There is a significant subset of real LEOs for which this is true. We address this subset to estimate $\Delta B/B$ and $\Delta\rho/\rho$ simultaneously. Parameter $\Delta B/B$ is modeled directly, and simply, as an exponentially correlated Gauss-Markov sequence. The modeling of parameter $\Delta\rho/\rho$ will now be described.

17.6 Atmospheric Density Errors

The atmospheric density error estimation model presented here was developed by Wright between October 1996 and May 1998.

A stationary exponential Gauss-Markov sequence for relative air-density error is modeled at mean perigee height. This auto-correlated sequence is driven by a white noise sequence whose variance is derived from a function that produced Figure 17.1, and from an input exponential half-life time constant. A map from perigee height to height at current time is derived from Figure 17.1 and used for each point on the spacecraft trajectory integrator time grid. It is important to note that the spacecraft physically samples its perigee height region, *where air-density is greatest*, once per orbit period. Thus there is a strong periodic connection between the height modeled by the Gauss-Markov sequence and spacecraft height. Filtered estimates of corrections to the CIRA 1972 air-density model will thus be dominated by tracking measurements in the perigee region. It is thus connected to air-density physics. CIRA 1972 is essentially a synonym for Jacchia 1971. It produces modeled atmospheric density from a

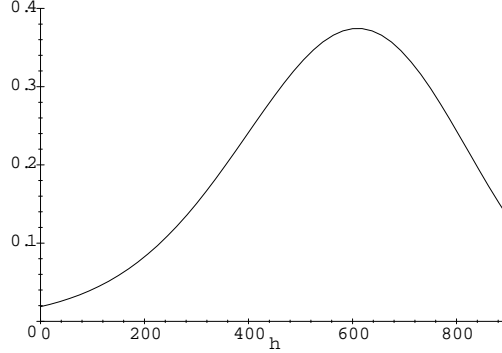


Figure 17.1: Sigma for Relative Error in Air Density

lower height of 110 km to an upper height of 2000 km. However, the analysis from which Figure 17.1 was derived extended to an upper height of only 800 km.

Two complete solar cycles of a_P and F_{10} measurements provide a base-line from which the variance on height dependent relative air-density error has been averaged. Let us denote associated average values for a_P and F_{10} with $\langle a_P \rangle$ and $\langle F_{10} \rangle$.

This is the first time this physically-connected air-drag error model will have been used in any system for operational orbit determination, or in the analysis of orbit determination performance.

17.6.1 Relative Error Base-Line Model on Air-Density

Define:

$$\sigma_{\Delta\rho/\bar{\rho}}(h) = \sqrt{E \left\{ \left(\frac{\Delta\rho(h)}{\bar{\rho}(h)} \right)^2 \right\}} = \frac{\sqrt{E \{ (\Delta\rho(h))^2 \}}}{\bar{\rho}(h)} = \frac{\sigma_{\Delta\rho}(h)}{\bar{\rho}(h)} \quad (17.25)$$

where:

$$\Delta\rho(h) = \rho(h) - \bar{\rho}(h) \quad (17.26)$$

where $\rho(h)$ is true atmospheric density at height h (in km), and $\bar{\rho}(h)$ is the associated value of estimated atmospheric density according to the CIRA 1972 atmospheric density model. An estimated graph of $\sigma_{\Delta\rho/\bar{\rho}}(h)$ is presented by Figure 17.1.

Let us denote this graph with the function:

$$f(h) = \sigma_{\Delta\rho/\bar{\rho}}(h) \quad (17.27)$$

17.6.2 Gauss-Markov Sequence on Air-Density Relative Error

A stationary exponential Gauss-Markov sequence was introduced in Section 13.1 as a stochastic sequence that satisfies Kalman's model Eqs. 4.1 and 4.2. *Stationary* means that the variance on this sequence does not change with time. We shall use this sequence to model the base-line relative error:

$$D(t) = \frac{\Delta\rho_{h_P}(t)}{\bar{\rho}_{h_P}(t)} \quad (17.28)$$

in atmospheric density at mean perigee height h_P , where $D(t)$ satisfies the equation:

$$D(t_{k+1}) = \Phi(t_{k+1}, t_k) D(t_k) + \sqrt{1 - \Phi^2(t_{k+1}, t_k)} w(t_k) \quad , \quad k \in \{0, 1, 2, \dots\} \quad (17.29)$$

where $w(t)$ is a Gaussian white random variable with mean zero and variance σ_w^2 , and where:

$$D(t_0) = w(t_0) \quad (17.30)$$

$$\Phi(t_{k+1}, t_k) = e^{\alpha(t_{k+1} - t_k)} \quad (17.31)$$

$$\text{constant } \alpha < 0 \quad (17.32)$$

Note from Figure 17.1 that the variance $\sigma_{\Delta\rho/\bar{\rho}}^2$ on air-density-relative-error varies significantly with height h . Thus it is appropriate to choose a height for $\Delta\rho/\bar{\rho}$ that is fixed, or approximately fixed. Therefore we have anchored the relative air-density error to mean perigee height h_P . Note that:

$$E\{D^2(t_k)\} = \sigma_w^2, \text{ for each } k, \quad (17.33)$$

that $D(t_k)$ has a Gaussian distribution, and that $D(t_k)$ is Markov.

17.6.3 Propagation of State Estimate

Let $\hat{D}_{n|m}$ denote an optimal estimate of $D(t_n)$, where t_n is the epoch for $\hat{D}_{n|m}$ and t_m is the time of last measurement. Then according to Sherman's Theorem Eq. 3.9:

$$\hat{D}_{n|m} = E\{D(t_n) | y_m\} \quad (17.34)$$

Apply Eq. 17.34 to Eq. 17.29, where y_k at time t_k was the last measurement processed:

$$\hat{D}_{k+1|k} = \Phi(t_{k+1}, t_k) \hat{D}_{k|k} \quad (17.35)$$

Eq. 17.35 is the filter state estimate propagation equation for the filter time-update. Given measurement y_{k+1} at time t_{k+1} use Kalman's filter measurement-update theorem, derived from application of Eq. 17.34 to $D(t_{k+1})$, for the representation:

$$\hat{D}_{k+1|k+1} = E \{ D(t_{k+1}) | y_{k+1} \} \quad (17.36)$$

Propagation of $\hat{D}_{k+1|k+1}$ to time t_{k+2} :

$$\hat{D}_{k+2|k+1} = \Phi(t_{k+2}, t_{k+1}) \hat{D}_{k+1|k+1}$$

17.6.4 State Estimate Error

Define the error in $\hat{D}_{n|m}$ by:

$$\delta \hat{D}_{n|m} = D_n - \hat{D}_{n|m} \quad (17.37)$$

Insert Eqs. 17.35 and 17.29 into Eq. 17.37:

$$\delta D_{k+1|k} = \Phi(t_{k+1}, t_k) \delta D_{k|k} + \sqrt{1 - \Phi_{k+1,k}^2} w_k \quad (17.38)$$

17.6.5 Process Noise Covariance

Base-Line Model

Square Eq. 17.38 and apply the expectation operator to get:

$$E \left\{ (\delta D_{k+1|k})^2 \right\} = \Phi^2(t_{k+1}, t_k) E \left\{ (\delta D_{k|k})^2 \right\} + (1 - \Phi_{k+1,k}^2) \sigma_w^2 \quad (17.39)$$

Notice that:

$$E \left\{ (\delta D_{k|k})^2 \right\} < \sigma_w^2 \quad (17.40)$$

due to the processing of measurements by the optimal filter. Thus the stochastic sequence defined by Eq. 17.38 is not stationary. The second term in the right-hand side of Eq. 17.39 is the base-line process noise covariance for deweighting prior estimates of D :

$$q_{k+1,k}^{\Delta\rho/\rho} = (1 - \Phi_{k+1,k}^2) \sigma_w^2 \quad (17.41)$$

For long propagation time intervals $[t_k, t_{k+1}]$ the factor $(1 - \Phi_{k+1,k}^2)$ tends to unity, and this adds σ_w^2 . For short propagation time intervals $[t_k, t_{k+1}]$ the factor

$(1 - \Phi_{k+1,k}^2)$ tends to zero, and only a small part of σ_w^2 is added. Thus the factor $(1 - \Phi_{k+1,k}^2)$ drives the variance $E \left\{ (\delta D_{k+1|k})^2 \right\}$ toward σ_w^2 in the absence of measurements, but adds little or nothing during dense measurements. When mean values $\langle a_P \rangle$ and $\langle F_{10} \rangle$ of a_P and F_{10} are experienced, this is appropriate for the base-line Gauss-Markov model.

But for an active Sun (solar maximum) when a_P and F_{10} are much larger than $\langle a_P \rangle$ and $\langle F_{10} \rangle$, then an important model extension is called for. The Gauss-Markov model must be immediately interrupted to open the filter gain, particularly during dense measurements.

Active Sun

The CIRA 1972 atmospheric density model $\bar{\rho}$ is a function of several arguments. It will suffice in this section to write:

$$\bar{\rho} = \bar{\rho}(h, F_{10}, \bar{F}_{10}, a_P, t_{k+1}) \quad (17.42)$$

Define the unitless ratio:

$$R = \frac{\bar{\rho}(h_P, F_{10}, \bar{F}_{10}, a_P, t_{k+1})}{\bar{\rho}(h_P, \langle F_{10} \rangle, \langle F_{10} \rangle, \langle a_P \rangle, t_{k+1})} \quad (17.43)$$

and define R_{max} . Initialize:

$$R_{max} = 1 \quad (17.44)$$

Algorithm: If $R > R_{max}$, set $R_{max} = R$ and define:

$$Q_{k+1,k}^{\Delta\rho/\rho} = R^2 \sigma_w^2 \quad (17.45)$$

else, set $R_{max} = R$ and define:

$$Q_{k+1,k}^{\Delta\rho/\rho} = q_{k+1,k}^{\Delta\rho/\rho} \quad (17.46)$$

Use $Q_{k+1,k}^{\Delta\rho/\rho}$ for air-density error process noise covariance deweighting. Effect: When $\bar{\rho}$ is increasing at h_P due to F_{10} , \bar{F}_{10} , and a_P , then the air density variance and filter gain are immediately opened to enable tracking measurements to estimate significant changes in air density. But when $\bar{\rho}$ is decreasing at h_P due to F_{10} , \bar{F}_{10} , and a_P , then the air density variance and filter gain begin their return to the baseline model.

The active Sun model is an innovation in that it enables real time range and/or Doppler spacecraft tracking measurements to estimate atmospheric density corrections due to extreme solar activity (*e.g.*, CME), or due to mean or quiet solar activity, in real time.

Real-Time Filtering of Tracking Data

Near real time values of F_{10} and a_P are derived from global observatory measurements. The 10 centimeter solar flux measurement F_{10} is available once per day in near real time. The mean value \bar{F}_{10} , averaged over the previous four solar rotations, is also available once per day. The geomagnetic measurement a_P , or K_P , is available in near real time at the end of each three hour interval.

Time propagations of F_{10} and a_P provide values for real-time filtering of LEO tracking measurements. There is an obvious problem here. Namely, the predicted values of F_{10} and a_P always suffer from propagation error. Typically, F_{10} and a_P are treated as random walk stochastic sequences. Thus the last observed value is used as the predicted estimate. Consider the prediction error in a_P during the geomagnetic response to a commencement of coronal mass ejection at solar maximum. The a_P may increase by several hundred per-cent at any time during any three-hour prediction interval. Atmospheric density is extremely sensitive to sudden changes in a_P . Since there exists no real-time measurement of a_P , it is impossible to explicitly model the sudden change in atmospheric density in real-time. What can one do?

If one forces the error variance on the relative atmospheric density parameter to remain wide open during the entire time interval of solar maximum (say from two to four years), then the filter will be sensitive to sudden changes in atmospheric density due to range and/or Doppler tracking data. However, filter orbit accuracy performance will suffer during much of this time interval due to passive geomagnetic activity and an excessively open gain on the relative atmospheric density parameter. Also, the atmospheric density parameter will experience huge and sudden estimate corrections, both positive and negative, while the filter distributes its over-corrections throughout the multi-dimensional state space with time.

If one controls the error variance on the relative atmospheric density parameter to behave in accordance with most variations in a_P , then the filter performance will be relatively satisfactory and smooth most of the time. However, the filter will not be responsive to sudden changes in atmospheric density.

Initialization

For filter initialization, define:

$$Q_{k+1,k}^{\Delta\rho/\rho} = R^2 \sigma_w^2 \quad (17.47)$$

The filter should be restarted, using the filter restart capability, following filter initialization.

17.6.6 Transform From Perigee Height to Current Height

Recall Eqs. 17.27 and 17.25 to write:

$$\frac{\sigma_{\Delta\rho}(h)}{\bar{\rho}(h)} = f(h) \frac{f(h_P)}{f(h_P)} = \frac{f(h)}{f(h_P)} f(h_P) = \frac{f(h)}{f(h_P)} \frac{\sigma_{\Delta\rho}(h_P)}{\bar{\rho}(h_P)} \quad (17.48)$$

That is:

$$D(h) = \frac{f(h)}{f(h_P)} D(h_P) \quad (17.49)$$

Eq. 17.48 derives from:

$$\frac{\Delta\rho(h)}{\bar{\rho}(h)} = \frac{f(h)}{f(h_P)} \frac{\Delta\rho(h_P)}{\bar{\rho}(h_P)} \quad (17.50)$$

That is, square Eq. 17.50, apply expectation, and take square root to get Eq. 17.48. From Eq. 17.26:

$$\begin{aligned} \rho(h) &= \bar{\rho}(h) + \Delta\rho(h) \\ &= \bar{\rho}(h) \left[1 + \frac{\Delta\rho(h)}{\bar{\rho}(h)} \right] \end{aligned} \quad (17.51)$$

Insert Eq. 17.50 into Eq. 17.51:

$$\rho(h) = \bar{\rho}(h) \left[1 + \frac{f(h)}{f(h_P)} \frac{\Delta\rho(h_P)}{\bar{\rho}(h_P)} \right]$$

and apply Sherman's Theorem to get:

$$\hat{\rho}(h) = \bar{\rho}(h) \left[1 + \frac{f(h)}{f(h_P)} \frac{\Delta\hat{\rho}(h_P)}{\bar{\rho}(h_P)} \right]$$

or:

$$\hat{\rho}(h) = \bar{\rho}(h) \left[1 + \frac{f(h)}{f(h_P)} \hat{D} \right] \quad (17.52)$$

Eq. 17.52 defines the method to map the filter estimate \hat{D} at perigee height h_P to the optimal estimate $\hat{\rho}(h)$ of atmospheric density, at current height h , for use in trajectory propagation.

17.6.7 Orbit Error Covariance

Acceleration Errors

Use the stochastic differential operator Δ on Eq. 17.22 to relate random errors in \ddot{z}_D to random errors in $\hat{\rho}$, where both are serially correlated:

$$\Delta\ddot{z}_D = -\frac{1}{2} \left(\frac{C_{DA}}{m} \right) \dot{s}^2 \Delta\hat{\rho}K \quad (17.53)$$

where $\Delta\rho$ is represented according to Eq. 17.50:

$$\Delta\rho(h) = \bar{\rho}(h) \frac{f(h)}{f(h_P)} \frac{\Delta\rho(h_P)}{\bar{\rho}(h_P)} = \bar{\rho}(h) \frac{f(h)}{f(h_P)} D \quad (17.54)$$

Orbit Errors

Let Z^i denote the 6×1 column matrix of position and velocity components, with epoch at t_i . From the theory of Variation of Parameters in Universal Variables:

$$\frac{dZ^i}{dt} = \frac{\partial Z^i}{\partial \dot{z}} \ddot{z}_D$$

or:

$$dZ^i = \frac{\partial Z^i}{\partial \dot{z}} \ddot{z}_D dt$$

where $\partial Z^i / \partial \dot{z}_D$ is a 6×3 matrix and \ddot{z}_D is a 3×1 matrix. Use the stochastic differential operator Δ on the equation above to relate random errors in Z^i to random errors in \ddot{z}_D , where both are serially correlated, and exchange differential operators on the left-hand side:

$$d(\Delta Z^i) = \frac{\partial Z^i}{\partial \dot{z}} \Delta \ddot{z}_D dt$$

Use the definite integral operator $\int_{t_i}^{t_j}$ on the equation above to get:

$$\Delta Z^i = \int_{t_i}^{t_j} \frac{\partial Z^i}{\partial \dot{z}} \Delta \ddot{z}_D dt$$

Apply the expectation operator $E\{\cdot\}$ to the outer product on the equation above to get:

$$E\{(\Delta Z^i)(\Delta Z^i)^T\} = \int_{t_i}^{t_j} \int_{t_i}^{t_j} \left(\frac{\partial Z^i}{\partial \dot{z}}(t) \right) G(t, \tau) \left(\frac{\partial Z^i}{\partial \dot{z}}(\tau) \right)^T dt d\tau \quad (17.55)$$

a 6×6 matrix, where:

$$G(t, \tau) = E\{\Delta \ddot{z}_D(t) \Delta \ddot{z}_D^T(\tau)\} \quad (17.56)$$

a 3×3 matrix, and it is assumed that:

$$E\{\Delta \ddot{z}_D\} = 0$$

$$E\{\Delta Z^i\} = 0.$$

Repeat the above development using conditional expectations $E\{\Delta Z^i|y\}$ and $E\{(\Delta Z^i)(\Delta Z^i)^T|y\}$ where y is an $m \times 1$ column matrix of time-ordered measurements. Invoke Sherman's Theorem to remove the optimal estimate $\Delta \hat{Z}^i$:

$$\Delta \hat{Z}^i = E \{ \Delta Z^i | y \}$$

so that:

$$E \{ \Delta Z^i | y \} - \Delta \hat{Z}^i = 0$$

It is in this sense that the above assumptions are made. The optimal filter will always remove $\Delta \hat{Z}^i$ so that the orbit error covariance $E \{ (\Delta Z^i) (\Delta Z^i)^T | y \}$ is always calculated about $\Delta \hat{Z}^i$, and not about the origin. Continue development with the implicit agreement that all expectations are conditioned on measurements of y .

Assume that the definite integral time $|t_j - t_i| < \epsilon$ is small. We will now make what will seem to be rather gross approximations. They are justified by the fact that we are looking for at most only two digits of significance in the calculation of state error covariance quantities. First notice for Variation of Parameters in Universal Variables that approximately:

$$\frac{\partial Z^i}{\partial \dot{z}} \cong \begin{bmatrix} 0_{3 \times 3} \\ I_{3 \times 3} \end{bmatrix}$$

Then:

$$\left(\frac{\partial Z^i}{\partial \dot{z}}(t) \right) G(t, \tau) \left(\frac{\partial Z^i}{\partial \dot{z}}(\tau) \right)^T \cong \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & G(t, \tau) \end{bmatrix}$$

Eq. 17.55 becomes:

$$E \{ (\Delta Z^i) (\Delta Z^i)^T \} \cong \int_{t_i}^{t_j} \int_{t_i}^{t_j} \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & G(t, \tau) \end{bmatrix} dt d\tau$$

Further, insert the double impulsive sifting integral into the integrand to get:

$$E \{ (\Delta Z^i) (\Delta Z^i)^T \} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & G(t_i, t_i) [t_j - t_i]^2 \end{bmatrix} \quad (17.57)$$

Insert Eqs. 17.53 and 17.54 into 17.56 to get:

$$G = \left(-\frac{1}{2} \left(\frac{C_D A}{m} \right) \dot{s}^2 \right)^2 E \{ (\Delta \hat{\rho})^2 \} K K^T \quad (17.58)$$

The orbit error transition function Φ will integrate the velocity error covariance $G(t_i, t_i) [t_j - t_i]^2$ into position error covariance with a lag of one time unit $[t_j - t_i]$. A reasonable time increment upper bound is given by:

$$|t_j - t_i| \leq \epsilon = P/50$$

where P is orbit period.

Chapter 18

Solar Photon Pressure

When spacecraft height is sufficiently low, the perturbative accelerations and modeling errors due to solar photon pressure are masked by the perturbative accelerations and modeling errors due to air-drag. Knowledge of this height is necessary for setting the force model for numerical trajectory propagation, and for the estimation of drag and/or solar pressure parameters. Musen (see Baker [4] page 188) put this height at 800 km, whereas Vokrouhlicky [112] put this height at 300 km.

Hujesak [40] presents a useful survey for identification of state-of-the-art solar pressure modeling.

18.1 Coordinate Frame for Solar Pressure

Stochastic *stationary* Gauss-Markov sequences are used for estimation of observable solar pressure parameters. It is desirable to select a coordinate frame that best supports stationarity. The spacecraft-Sun line is a natural accessible choice. To this end, let \mathbf{r} denote the ECI position vector of the spacecraft center of mass, let \mathbf{S} denote the ECI position vector of the Sun center of mass, and let $\boldsymbol{\rho}$ denote the vector from Sun center of mass to spacecraft center of mass. Then:

$$\boldsymbol{\rho} = \mathbf{r} - \mathbf{S} \quad (18.1)$$

enables definition of the unit vector \mathbf{s}_1 from Sun to spacecraft:

$$\mathbf{s}_1 = \boldsymbol{\rho} / \rho = [\mathbf{i}]^T \begin{bmatrix} s_{11} \\ s_{12} \\ s_{13} \end{bmatrix} \quad (18.2)$$

where:

$$\rho = \sqrt{\boldsymbol{\rho} \cdot \boldsymbol{\rho}} \quad (18.3)$$

Use the orbit angular momentum defined in Section 1.2.1 to specify the unit vector \mathbf{W} . Then construct an intermediate vector \mathbf{T} orthogonal to both \mathbf{s}_1 and \mathbf{W} :

$$\mathbf{T} = \mathbf{W} \times \mathbf{s}_1 \quad (18.4)$$

and use its unit vector for the orthonormal basis:

$$\mathbf{s}_2 = \mathbf{T}/T = [\mathbf{i}]^T \begin{bmatrix} s_{21} \\ s_{22} \\ s_{23} \end{bmatrix} \quad (18.5)$$

where:

$$T = \sqrt{\mathbf{T} \cdot \mathbf{T}} \quad (18.6)$$

Finally:

$$\mathbf{s}_3 = \mathbf{s}_1 \times \mathbf{s}_2 = [\mathbf{i}]^T \begin{bmatrix} s_{31} \\ s_{32} \\ s_{33} \end{bmatrix} \quad (18.7)$$

completes the orthonormal vector basis $[\mathbf{s}]$ for specification of solar pressure:

$$[\mathbf{s}] = \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \mathbf{s}_3 \end{bmatrix} \quad (18.8)$$

The direction of solar photon motion is defined by \mathbf{s}_1 . The rotation R_{si} from $[\mathbf{i}]$ to $[\mathbf{s}]$ is given by:

$$[\mathbf{s}] = R_{si} [\mathbf{i}] \quad (18.9)$$

where:

$$R_{si} = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix} \quad (18.10)$$

We also have need of unit vector \mathbf{U} , defined in Section 1.2.1, to specify the direction of IR photons from Earth to spacecraft.

18.2 Solar Pressure Acceleration

Pechevick's model[95] is implemented for two approximations to the satellite shape and characteristics. The difference between the two models is simply a multiplicative scale factor which will be absorbed during the estimation process. The sphere with perfect absorption model is the model most commonly used in trajectory generation software.

18.2.1 Notation

Symbols used for our solar pressure acceleration model are identified in the following table:

m	mass of spacecraft
L	luminosity of Sun
ρ	distance from Sun to spacecraft
$\ddot{\mathbf{r}}_S$	solar pressure acceleration of spacecraft
r_D	diffuse reflection coefficient
r_S	specular reflection coefficient
$\hat{\mathbf{n}}$	unit outward normal vector to spacecraft surface element
$\hat{\mathbf{k}}$	unit vector defines direction of motion of photons from Sun to spacecraft
dA	infinitesimal element of spacecraft surface area
c	speed of light

18.2.2 Conservation of Linear Momentum

From conservation of linear momentum, Pechenick[95] integrates over a convex illuminated spacecraft surface to find the following expression for spacecraft solar pressure acceleration:

$$\ddot{\mathbf{r}}_S = \frac{L}{4\pi\rho^2mc} \iint \left[(1 - r_S) \hat{\mathbf{k}} - \frac{2}{3} r_D \hat{\mathbf{n}} - 2r_S (-\hat{\mathbf{k}} \cdot \hat{\mathbf{n}}) \hat{\mathbf{n}} \right] (-\hat{\mathbf{k}} \cdot \hat{\mathbf{n}}) dA \quad (18.11)$$

where:

$$\hat{\mathbf{k}} \cdot \hat{\mathbf{n}} < 0 \quad (18.12)$$

$$\hat{\mathbf{k}} = \mathbf{s}_1 \quad (18.13)$$

$$L = 3.8530 \times 10^{26} \frac{\text{joules}}{\text{sec}} \left(\frac{\text{kg m}^2}{\text{sec}^3} \right) \quad (18.14)$$

$$c = 2.99792458 \times 10^8 \frac{\text{m}}{\text{sec}} \quad (18.15)$$

$$1 \text{ au} = 1.49600 \times 10^{11} \text{ m} \quad (18.16)$$

From Eq. 18.11 Pechenick derives the solar pressure acceleration for a spherical surface, a flat plate, a cylinder with closed ends, and various surface orientation averages over the flat plate and cylinder.

18.2.3 Spherical Surface Diffuse Reflection

Spacecraft solar pressure acceleration due to a spherical surface:

$$\ddot{\mathbf{r}}_S = \frac{L}{4\pi\rho^2mc} \pi R_s^2 \left(1 + \frac{4}{9}\right) r_D \hat{\mathbf{k}} \quad (18.17)$$

where R_s is radius of the sphere. Define a constant:

$$K_S = \frac{\left(1 + \frac{4}{9}\right)L}{4c}, \quad \left(\text{default: } 4.6411 \times 10^{17} \frac{\text{kg}\cdot\text{m}}{\text{sec}^2}\right) \quad (18.18)$$

This leaves r_D , ρ , m , and R_s for use as time variables. Variations in ρ can be as large as twice the orbit semi-major axis if the orbit line of nodes lies along the Sun-spacecraft line. Then:

$$\ddot{\mathbf{r}}_S = K_S \left(\frac{R_s^2}{\rho^2 m}\right) r_D \hat{\mathbf{k}} \quad (18.19)$$

Example: For a spherical spacecraft with $\rho = 1$ au, $r_D = 0.75$, $R_s = 1$ m, and $m = 1$ kg we have:

$$|\ddot{\mathbf{r}}_S| = 1.5553 \times 10^{-5} \frac{\text{m}}{\text{sec}^2}$$

Eq. 18.17 can be written in conventional operational form for diffuse photon reflection:

$$\ddot{\mathbf{r}}_S = \frac{L}{4\pi\rho^2c} \frac{A_S}{m} \left(1 + \frac{4}{9}\right) C_R \hat{\mathbf{k}} \quad (18.20)$$

where C_R is the solar pressure coefficient and A_S is the spacecraft area exposed to the Sun:

$$C_R = r_D \text{ and } A_S = \pi R_s^2 \quad (18.21)$$

18.2.4 Spherical Surface Perfect Absorption

For perfect photon absorption:

$$\ddot{\mathbf{r}}_S = \frac{L}{4\pi\rho^2c} \frac{A_S}{m} C_R \hat{\mathbf{k}} \quad (18.22)$$

18.3 Eclipse Modeling

18.3.1 Selection

The cylindrical model used by GSFC/GTDS for solar pressure eclipsing is rejected as a result of complete failure (10 sigma) of rigorous filter-smoother consistency testing with real GEO tracking data.

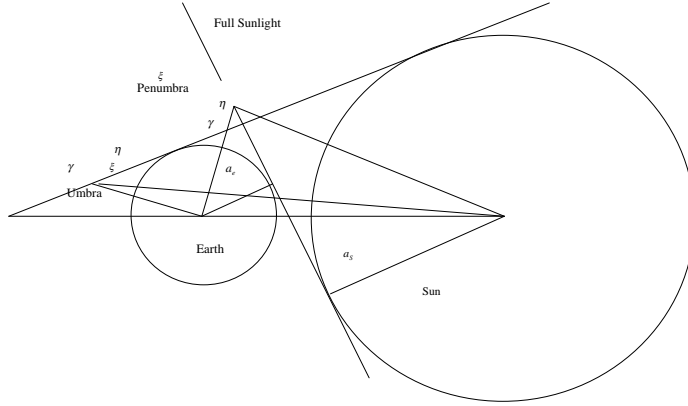


Figure 18.1: Dual-Cone Eclipse Geometry

Computational tractability currently excludes use of an atmospheric refraction model for solar pressure eclipsing. For example, consider Vokrouhlicky's [112] atmospheric refraction model. The angular width of a 50 km atmospheric slab, when seen from LAGEOS' height of 5500 km, is about 16 arcmin, and this is half the apparent size of the undistorted solar disk. Due to differential atmospheric refraction of solar rays, the entire solar disk is easily contained within the 16 arcmin atmospheric width. The apparent solar image is highly flattened. The associated radiant force is highly correlated with this refraction effect, and the two-cone umbra-penumbra definition is made obsolete. Atmospheric refraction produces a long-lived perturbation tail of solar rays deep into the umbra region for the two-cone model.

Until a tractable atmospheric refraction model is available, we shall use Baker's [4] two-cone eclipsing model with three sharply defined geometrical conditions: full sunlight, penumbra (partial eclipse), and umbra (total eclipse). Physically, the two-cone model is appropriate for a planet with no atmosphere. Baker's model ignores atmospheric refraction and solar limb darkening.

Rigorous filter-smoother consistency testing with real GEO tracking data demonstrated performance just within 3 sigmas. While poor, this result is far superior to that using a cylindrical eclipsing model. Also, the filter-smoother test result confirms Vokrouhlicky's analysis results relating to absence of models for atmospheric refraction and solar limb darkening.

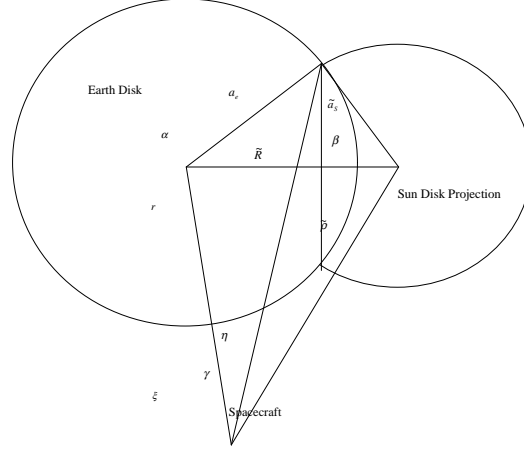


Figure 18.2: Baker's Solar Obscura

18.3.2 Baker's Dual-Cone Eclipse Model

Baker [4] defines:

$$f(\gamma, \eta, \xi) = \begin{cases} 1 & \xi > \gamma + \eta & \text{full sunlight} \\ 0 & \eta + \xi \leq \gamma & \text{umbra, none} \\ f_1 & \eta + \xi > \gamma & \text{penumbra} \\ f_2 & \gamma + \xi < \eta & \text{annular eclipse} \end{cases} \quad (18.23)$$

where:

$$f_1 = 1 - \left[\left(\frac{\gamma}{\eta} \right)^2 \left(\alpha - \frac{1}{2} \sin(2\alpha) \right) + \left(\beta - \frac{1}{2} \sin(2\beta) \right) \right] / \pi \quad (18.24)$$

$$f_2 = \left(\frac{\gamma}{\eta} \right)^2 \quad (18.25)$$

$$\sin \gamma = a_e / r \quad (18.26)$$

$$\sin \eta = a_s / \rho \quad (18.27)$$

$$\cos \xi = \mathbf{r} \cdot \boldsymbol{\rho} \quad (18.28)$$

$$\sin \alpha = \frac{2K}{\xi\gamma} \quad (18.29)$$

$$\sin \beta = \frac{2K}{\xi\eta} \quad (18.30)$$

$$\cos \alpha = \frac{\xi^2 + \gamma^2 - \eta^2}{2\xi\gamma} \quad (18.31)$$

$$\cos \beta = \frac{\xi^2 - \gamma^2 + \eta^2}{2\xi\eta} \quad (18.32)$$

$$S = \frac{\xi + \gamma + \eta}{2} \quad (18.33)$$

$$K = \sqrt{S(S - \eta)(S - \gamma)(S - \xi)} \quad (18.34)$$

$$a_e = 1 \text{ er} \simeq r\gamma \quad (18.35)$$

$$\tilde{a}_S = 108.966 \text{ er} \simeq r\eta \quad (18.36)$$

$$\tilde{R} \simeq r\xi \quad (18.37)$$

18.3.3 Earth Radius

The errors incurred by use of a two-cone model can be reduced by artificially inflating the modeled Earth radius so as to increase its shadow size. See Vokrouhlicky [112], midway through Section 5. Hujsak [40] cites McCarthy [71] and Link [63] on this practice, and notes that Link traces it back to 1707. McCarthy recommends increasing the effective radius of the Earth by 24 km to 6402 km, with 1738 km for radius of the Moon, and 696000 km for radius of the Sun.

18.4 Stochastic Solar Pressure Error Model

Let p denote a unitless time-varying Gauss-Markov parameter for solar pressure defined by:

$$p = r_D - \bar{r}_D \quad (18.38)$$

where r_D is associated with Eq. 18.17, and \bar{r}_D denotes a model constant with default value $\bar{r}_D = 0.75$. Insert Eq. 18.38 into Eq. 18.19 to get:

$$\ddot{\mathbf{r}}_S = K_S \left(\frac{R_s^2}{\rho^2 m} \right) (\bar{r}_D + p(t)) \hat{\mathbf{k}} \quad (18.39)$$

where $p(t)$ is a stationary Gauss-Markov sequence:

$$p(t_{k+1}) = \Phi(t_{k+1}, t_k)p(t_k) + \sqrt{1 - \Phi^2(t_{k+1}, t_k)}w(t_k) \quad , \quad k \in \{0, 1, 2, \dots\} \quad (18.40)$$

defined by Eqs. 17.29 (with $D \rightarrow p$) through 17.40, except that solar pressure values are assigned to σ_ω^2 and α .

Chapter 19

GPS Solar Pressure Models

19.1 GSPM.04a

Implementation of the JPL GSPM.04a models for Block IIA and Block IIR satellites follows Bar-Sever and Kuang (2004) [6]. This model was derived from a study of non-eclipsing satellites. The model is dependent on the angle of the sun out of the orbit plane (angle β), but holds the absolute value of the angle β at 14.5 degrees for cases where β goes below 14.5, as is the case for eclipsing satellites. The form of the model is the same for Block IIA and Block IIR satellites, but the coefficients are different.

19.2 GSPM.04ae

Implementation of the JPL GSPM.04ae models for Block IIA and Block IIR satellites follows reference Bar-Sever and Kuang (2005) [7]. This model was derived as an extension of the GSPM.04a models Bar-Sever and Kuang (2004) [6], where the models deviate from the GSPM.04a models only for eclipsing satellites. The model is dependent on the angle β of the sun out of the orbit plane, but drops one term involving division by the sine of β when β goes below one degree. The form of the model is the same for Block IIA and Block IIR satellites, but the coefficients are different.

19.3 AeroT20

Implementation of the Aerospace T20 model for use with Block IIA satellites follows O'Toole[91]. This model originally presented in Bar-Sever and Kuang (2004) [6].

19.4 AeroT30

Implementation of the Aerospace T30 model for use with Block IIA satellites follows O'Toole[91]. This model originally presented in Fliegel [24].

19.5 Estimation

Consider estimation of the K_1 and K_2 parameters of the GPS solar pressure model. K_1 is a unitless parameter which scales the acceleration along the sun to satellite line. The nominal value of K_1 is 1.0. K_2 is provided in units of 10^{-12}m/s^2 and specifies the acceleration along the GPS body-fixed Y axis. This is often referred to as the Y-bias acceleration. The Y axis points in opposite directions for Block IIA and Block IIR satellites when they are in their nominal attitudes. Nominal values for K_2 are on the order of 1. For estimation purposes, the complete values of K_1 and K_2 are given as

$$K_1 = \bar{K}_1 + K_{1,GM} \quad (19.1)$$

$$K_2 = \bar{K}_2 + K_{2,GM} \quad (19.2)$$

where \bar{K}_1 and \bar{K}_2 are constant nominal values and $K_{1,GM}$ and $K_{2,GM}$ are sequentially correlated Gauss-Markov random variables.

Chapter 20

Spacecraft Thrusting

The Frenét vector basis $[\mathbf{f}] = (\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3)^T$ was selected for definition of coordinate frame to represent input thrust accelerations. Many operational thrust maneuvers are specified to change only the orbit period (or semi-major axis) with thrusting along the spacecraft inertial velocity vector. The second vector component \mathbf{f}_2 of the Frenét basis isolates this effect. See Eq. 1.13 through Eq. 1.20 for complete definition of the Frenét vector basis.

Impulsive maneuver models are physically unrealistic because the thrust burn time interval is zero – infinitely small. Physically realistic maneuver modeling imitates the maneuver with an appropriate finite thrust burn time – for *finite maneuver*.

20.1 Impulsive Maneuver Model

The impulsive acceleration model is always an approximation. It is a good approximation when the thrusting time interval is very short compared to orbit period. When the thrusting time interval is significant compared to orbit period, the impulsive acceleration model is essentially useless.

Model the impulsive acceleration as a three-dimensional vector Dirac delta functional $\mathbf{A}_D(t)$. Then in the three-dimensional velocity space we have a (discontinuous) Heaviside step functional vector $\Delta\mathbf{V}(t)$, and in the three-dimensional position space we have a (continuous) corner functional vector. $\Delta\mathbf{V}(t)$ is a generalized integral to $\mathbf{A}_D(t)$.

20.1.1 Trajectory

Define components of vector $\Delta\mathbf{V} = \Delta\mathbf{V}(t)$ with the equation:

$$\Delta\mathbf{V} = [\mathbf{i}]^T \Delta V^i = [\mathbf{f}]^T \Delta V^f \quad (20.1)$$

where:

$$\Delta V^i = \begin{bmatrix} \Delta V_1^i \\ \Delta V_2^i \\ \Delta V_3^i \end{bmatrix}, \text{ inertial components} \quad (20.2)$$

$$\Delta V^f = \begin{bmatrix} \Delta V_1^f \\ \Delta V_2^f \\ \Delta V_3^f \end{bmatrix}, \text{ Frenét components} \quad (20.3)$$

Insert Eq. 1.20 into Eq. 20.1:

$$\Delta \mathbf{V} = [\mathbf{i}]^T \Delta V^i = [\mathbf{i}]^T R_{fi}^T \Delta V^f \quad (20.4)$$

to get:

$$\Delta V^i = R_{fi}^T \Delta V^f \quad (20.5)$$

$$\Delta V^f = R_{fi} \Delta V^i \quad (20.6)$$

Eq. 20.5 enables conversion of input Frenét components of ΔV^f to inertial components of ΔV^i . The inertial components of $\Delta V^i = \Delta V^i(t_C)$ are added directly to velocity components of the predicted trajectory estimate at time of centroid t_C .

20.1.2 Trajectory Error Covariance

Let us denote the error in vector $\Delta \mathbf{V}$ with $\delta \mathbf{V}$. Replace Δ with δ in the notation above to get:

$$\delta V^i = R_{fi}^T \delta V^f \quad (20.7)$$

$$\delta V^f = R_{fi} \delta V^i \quad (20.8)$$

Take the expectation of the outer product on Eq. 20.7:

$$E \left\{ \delta V^i (\delta V^i)^T \right\} = R_{fi}^T E \left\{ \delta V^f (\delta V^f)^T \right\} R_{fi} \quad (20.9)$$

and give names to the 3×3 covariance matrices:

$$P_{\delta V}^i = E \left\{ \delta V^i (\delta V^i)^T \right\} \quad (20.10)$$

$$P_{\delta V}^f = E \left\{ \delta V^f (\delta V^f)^T \right\} \quad (20.11)$$

Then:

$$P_{\delta V}^i = R_{fi}^T P_{\delta V}^f R_{fi} \quad (20.12)$$

The user will specify the square-roots (sigmas) to the diagonal elements of the input Frenét covariance matrix $P_{\delta V}^f$ in user units, the program will convert to internal units and square the user inputs to form variances, insert the variances into the diagonal elements of $P_{\delta V}^f$, and put zeros in the off-diagonal elements of $P_{\delta V}^f$. The program will then evaluate R_{fi} and perform the multiplications indicated by Eq. 20.12 to calculate $P_{\delta V}^i$. The program will add $P_{\delta V}^i = P_{\delta V}^i(t_C)$ to the velocity covariance submatrix at time t_C .

20.1.3 Time of Centroid t_C

Let t_B and t_E denote begin-burn and end-burn times, and let $A^f(t)$ denote a Frenét matrix 3×1 acceleration function across $[t_B, t_E]$. The time of centroid for each Frenét component t_{Cj} is defined by:

$$t_{Cj} = \frac{M_j^f}{\Delta V_j^f}, \quad j \in \{1, 2, 3\} \quad (20.13)$$

where:

$$M_j^f = \int_{t_B}^{t_E} t A_j^f(t) dt \quad (20.14)$$

$$\Delta V_j^f = \int_{t_B}^{t_E} A_j^f(t) dt \quad (20.15)$$

If the acceleration is given along Frenét component k only, where $k \in \{1, 2, 3\}$, then:

$$t_C = t_{Ck} \quad (20.16)$$

If not, then a weighted combination of times is in order:

$$t_C = \sum_{k=1}^3 W_k t_{Ck} \quad (20.17)$$

where:

$$\sum_{k=1}^3 W_k = 1, \quad W_k \geq 0 \quad (20.18)$$

and where weight W_k , $k \in \{1, 2, 3\}$, is derived from $|\Delta V_j^f|$, $j \in \{1, 2, 3\}$; *e.g.*:

$$W_k = \frac{|\Delta V_k^f|}{\sum_{j=1}^3 |\Delta V_j^f|} \quad (20.19)$$

Constant Acceleration Components

Given that $A_j^f(t)$ is a time constant, then Eqs. 20.15, 20.14, and 20.13 are reduced to:

$$\Delta V_j^f = A_j^f (t_E - t_B) \quad (20.20)$$

$$M_j^f = \frac{1}{2} A_j^f (t_E^2 - t_B^2) \quad (20.21)$$

$$t_{Cj} = \frac{1}{2} (t_E + t_B) \quad (20.22)$$

20.2 Impulsive Maneuver Error Covariance

20.2.1 Notation

t_{k-1} , time of last measurement before burn

t_c , burn time centroid

$t_k = t_c^{(-)} = t_c - \epsilon/2$, start burn time

$t_{k+1} = t_c^{(+)} = t_c + \epsilon/2$ = end burn time

t_{k+2} = time of first measurement after burn

$\hat{P}_c^{\Delta V}$, cov for ΔV at t_c

20.2.2 Time Relations

$$t_{k-1} < t_k < t_c < t_{k+1} < t_{k+2}$$

$$\epsilon > 0$$

20.2.3 Covariance

Filter Running Forward with Time

Given $\hat{P}_{c|k-1}$ at time of last measurement before burn, then:

$$\hat{P}_{c|k-1}^{(-)} = \Phi_{c,k-1} \hat{P}_{c|k-1} \Phi_{c,k-1}^T + P_{c,k-1}^f \quad (20.23)$$

$$\hat{P}_{c|k-1}^{(+)} = \hat{P}_{c|k-1}^{(-)} + \hat{P}_c^{\Delta V} \quad (20.24)$$

$$\hat{P}_{k+2|k-1} = \Phi_{k+2,c} \hat{P}_{c|k-1}^{(+)} \Phi_{k+2,c}^T + P_{k+2,c}^f \quad (20.25)$$

Smoother Running Backward with Time

Nominally:

$$\tilde{P}_{L|L} = \hat{P}_{L|L} \quad (20.26)$$

⋮

$$\tilde{P}_{k+2|L} = \hat{P}_{k+2|k+2} + A_{k+2,k+3} \left[\tilde{P}_{k+3|L} - \hat{P}_{k+3|k+2} \right] A_{k+2,k+3}^T \quad (20.27)$$

The smoother covariance $\tilde{P}_{k+2|L}$ is now available at time of first measurement after burn. Then:

$$\tilde{P}_{c|L}^{(+)} = \hat{P}_{c|k-1}^{(+)} + A_{c,k+2} \left[\tilde{P}_{k+2|L} - \hat{P}_{k+2|k-1} \right] A_{c,k+2}^T \quad (20.28)$$

where:

$$A_{c,k+2} = \hat{P}_{c|k-1}^{(+)} \Phi_{k+2,c}^T \hat{P}_{k+2|k-1}^{-1} \quad (20.29)$$

and:

$$\tilde{P}_{c|L}^{(-)} = \hat{P}_{c|k-1}^{(-)} + A_{c,c} \left[\tilde{P}_{c|L}^{(+)} - \hat{P}_{c|k-1}^{(+)} \right] A_{c,c}^T \quad (20.30)$$

where:

$$A_{c,c} = \hat{P}_{c|k-1}^{(-)} \left(\hat{P}_{c|k-1}^{(+)} \right)^{-1} \quad (20.31)$$

20.3 Impulsive Maneuver Covariance

From the smoothed estimates $\tilde{X}_{k+1|L} = \tilde{X}_{c|L}^{(+)}$ and $\tilde{X}_{k|L} = \tilde{X}_{c|L}^{(-)}$, the difference $D\tilde{X}_{c|L} = \tilde{X}_{k+1|L} - \tilde{X}_{k|L}$ is calculated so as to quantify the difference in smoothed estimates, before and after the impulsive maneuver. The purpose here is to derive the associated error covariance $\tilde{P}_{c|L}$ defined by:

$$\tilde{P}_{c|L} = E \left\{ \delta D\tilde{X}_{c|L} \left(\delta D\tilde{X}_{c|L} \right)^T \right\} \quad (20.32)$$

on the error difference:

$$\delta D\tilde{X}_{c|L} = \delta\tilde{X}_{k+1|L} - \delta\tilde{X}_{k|L} \quad (20.33)$$

So:

$$\begin{aligned} \tilde{P}_{c|L} &= E \left\{ \left(\delta\tilde{X}_{k+1|L} - \delta\tilde{X}_{k|L} \right) \left(\delta\tilde{X}_{k+1|L} - \delta\tilde{X}_{k|L} \right)^T \right\} \\ &= E \left\{ \delta\tilde{X}_{k+1|L} \left(\delta\tilde{X}_{k+1|L} \right)^T \right\} + E \left\{ \delta\tilde{X}_{k|L} \left(\delta\tilde{X}_{k|L} \right)^T \right\} \\ &\quad - \left[E \left\{ \delta\tilde{X}_{k+1|L} \left(\delta\tilde{X}_{k|L} \right)^T \right\} + E \left\{ \delta\tilde{X}_{k|L} \left(\delta\tilde{X}_{k+1|L} \right)^T \right\} \right] \end{aligned} \quad (20.34)$$

Note that the sum in square brackets is symmetric, but neither of its terms is symmetric. Covariance matrices $E \left\{ \delta\tilde{X}_{k+1|L} \left(\delta\tilde{X}_{k+1|L} \right)^T \right\}$ and $E \left\{ \delta\tilde{X}_{k|L} \left(\delta\tilde{X}_{k|L} \right)^T \right\}$ are readily available from standard smoother output. But:

$$\tilde{P}_{k+1,k|L} = E \left\{ \delta\tilde{X}_{k+1|L} \left(\delta\tilde{X}_{k|L} \right)^T \right\} \quad (20.35)$$

and:

$$\tilde{P}_{k,k+1|L} = E \left\{ \delta\tilde{X}_{k|L} \left(\delta\tilde{X}_{k+1|L} \right)^T \right\} \quad (20.36)$$

are not available. Our purpose now is to derive a structure to evaluate $\tilde{P}_{k+1,k|L}$ and $\tilde{P}_{k,k+1|L}$. Begin with Meditch[77], page 221, Eq. 6.48:

$$\delta\tilde{X}_{k|L} = \delta\hat{X}_{k|k} - A_{k,k+1} \left(\tilde{X}_{k+1|L} - \hat{X}_{k+1|k} \right) \quad (20.37)$$

where:

$$A_{k,k+1} = \hat{P}_{k|k} \Phi_{k+1,k}^T \hat{P}_{k+1|k}^{-1} \quad (20.38)$$

Let X_k denote truth, and let $X_{k|j}$ denote an estimate. The error $\delta X_{k|j}$ in $X_{k|j}$ is defined by:

$$\delta X_{k|j} = X_k - X_{k|j} \quad (20.39)$$

Eq. 20.37 then becomes:

$$\delta \tilde{X}_{k|L} = \delta \hat{X}_{k|k} - A_{k,k+1} \left(\delta \hat{X}_{k+1|k} - \delta \tilde{X}_{k+1|L} \right) \quad (20.40)$$

Insert Eq. 20.40 twice into Eq. 20.36 to get:

$$\begin{aligned} \tilde{P}_{k,k+1|L} &= E \left\{ \delta \tilde{X}_{k|L} \left(\delta \tilde{X}_{k+1|L} \right)^T \right\} = E \left\{ \delta \hat{X}_{k|k} \left(\delta \hat{X}_{k+1|k+1} \right)^T \right\} \\ &\quad + A_{k,k+1} [B - C] (A_{k+1,k+2})^T \\ &\quad + \left(E \left\{ \delta \hat{X}_{k|k} \left(\delta \tilde{X}_{k+2|L} \right)^T \right\} - E \left\{ \delta \hat{X}_{k|k} \left(\delta \hat{X}_{k+2|k+1} \right)^T \right\} \right) (A_{k+1,k+2})^T \\ &\quad + A_{k,k+1} \left(E \left\{ \delta \tilde{X}_{k+1|N} \left(\delta \hat{X}_{k+1|k+1} \right)^T \right\} - \delta \hat{X}_{k+1|k+1} \left(\delta \hat{X}_{k+2|k+1} \right)^T \right) \end{aligned} \quad (20.41)$$

where:

$$B = E \left\{ \delta \hat{X}_{k+1|k} \left(\delta \hat{X}_{k+2|k+1} \right)^T \right\} + E \left\{ \delta \tilde{X}_{k+1|L} \left(\delta \tilde{X}_{k+2|L} \right)^T \right\} \quad (20.42)$$

$$C = E \left\{ \delta \tilde{X}_{k+1|L} \left(\delta \hat{X}_{k+2|k+1} \right)^T \right\} + E \left\{ \delta \hat{X}_{k+1|k} \left(\delta \tilde{X}_{k+2|L} \right)^T \right\} \quad (20.43)$$

20.3.1 Discussion

Eq. 20.41 appears to be recursive, containing both $E \left\{ \delta \tilde{X}_{k|L} \left(\delta \tilde{X}_{k+1|L} \right)^T \right\}$ and $E \left\{ \delta \tilde{X}_{k+1|L} \left(\delta \tilde{X}_{k+2|L} \right)^T \right\}$. If simplification can be found to remove the implied recursion, then implementation is feasible at a later date. If not, a conservative approximation to $\tilde{P}_{c|L}$ is given by:

$$\tilde{P}_{c|L} = E \left\{ \delta \tilde{X}_{k+1|L} \left(\delta \tilde{X}_{k+1|L} \right)^T \right\} + E \left\{ \delta \tilde{X}_{k|L} \left(\delta \tilde{X}_{k|L} \right)^T \right\}$$

where:

$$E \left\{ \delta \tilde{X}_{k+1|L} \left(\delta \tilde{X}_{k|L} \right)^T \right\} + E \left\{ \delta \tilde{X}_{k|L} \left(\delta \tilde{X}_{k+1|L} \right)^T \right\}$$

is ignored (set to zero). The recursion would need to be initialized from initial conditions in the filter and initial conditions in the smoother followed by associated sequential calculations.

20.4 Finite Maneuver Model

20.4.1 Kinematics

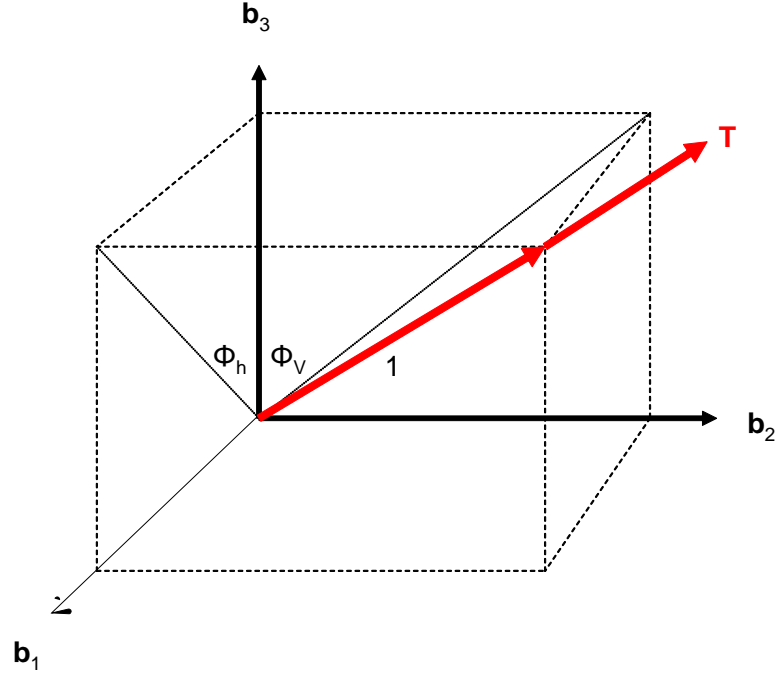


Figure 20.1: Unit Thrust Referred to Maneuver Frame

Let $[\mathbf{t}] = [\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3]^T$ denote an orthonormal vector basis variable $[\mathbf{t}] \in \{[\mathbf{i}], [\mathbf{u}], [\mathbf{f}]\}$ that may be assigned inertial $[\mathbf{i}]$, Gaussian $[\mathbf{u}]$, or Frenét $[\mathbf{f}]$ according to user selection. Thus $[\mathbf{t}]$ is a trajectory frame. Let $[\mathbf{b}] = [\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3]^T$ denote an orthonormal vector basis where \mathbf{b}_3 is defined to be the nominal thrust direction. Thus $[\mathbf{b}]$ defines the nominal maneuver (thrust direction) coordinate frame. Relate $[\mathbf{b}]$ to $[\mathbf{t}]$ with the 3×3 orthogonal rotation matrix R_{bt} :

$$[\mathbf{b}] = R_{bt} [\mathbf{t}] \quad (20.44)$$

$$[\mathbf{t}] = R_{bt}^T [\mathbf{b}] \quad (20.45)$$

When $R_{bt} = I_{3 \times 3}$, then the nominal thrust direction $\mathbf{b}_3 = \mathbf{t}_3$ is referred directly to a trajectory frame $[\mathbf{t}]$. Let $\mathbf{T} = \mathbf{T}(t)$ denote the time-dependent spacecraft thrust vector, and let $\hat{\mathbf{T}}$ denote the thrust unit vector:

$$\hat{\mathbf{T}} = \mathbf{T}/T \quad (20.46)$$

where thrust magnitude T is defined:

$$T = \sqrt{\mathbf{T} \cdot \mathbf{T}} \quad (20.47)$$

Then $\hat{\mathbf{T}}$ has components referred to $[\mathbf{b}]$:

$$\hat{\mathbf{T}} = [\mathbf{b}]^T \begin{bmatrix} \hat{T}_1 \\ \hat{T}_2 \\ \hat{T}_3 \end{bmatrix} \quad (20.48)$$

defined by horizontal angle φ_h and vertical angle φ_v such that:

$$\tan \varphi_h = \frac{\hat{T}_1}{\hat{T}_3} \quad (20.49)$$

$$\tan \varphi_v = \frac{\hat{T}_2}{\hat{T}_3} \quad (20.50)$$

$$\hat{T}_3 = (1 + \tan^2 \varphi_h + \tan^2 \varphi_v)^{-1/2} \quad (20.51)$$

By inspection:

$$\left(\hat{T}_1\right)^2 + \left(\hat{T}_2\right)^2 + \left(\hat{T}_3\right)^2 = 1$$

20.4.2 Dynamics

Let us denote the acceleration vector due to thrust with \mathbf{A}_T . Then:

$$\mathbf{A}_T = A_T \hat{\mathbf{T}} \quad (20.52)$$

where A_T is the acceleration magnitude. From Newton's second law:

$$T = mA_T \quad (20.53)$$

where m is spacecraft mass.

20.4.3 Stochastic Sequences

Angles φ_h and φ_v , and relative thrust magnitude $\Delta T/T$ are modeled as Gauss-Markov biases, and are not expected to be large.

Estimate	σ	States
none	N/A	none
T	$\varphi_h, \varphi_v, \Delta T/T$	$\Delta T/T$
T	φ_h, φ_v, T	T
φ_h, φ_v, T	$\varphi_h, \varphi_v, \Delta T/T$	$\varphi_h, \varphi_v, \Delta T/T$
φ_h, φ_v, T	φ_h, φ_v, T	φ_h, φ_v, T
φ_h, φ_v, T	T components	T components

Table 20.1: Table Caption

20.4.4 Estimation

Thrust magnitude T , or relative thrust magnitude $\Delta T/T$, angles φ_h and φ_v , and \mathbf{A}_T components on inertial, Gaussian, and Frenét frames may be estimated, with a variety of user selected options.

Part IV

Spacecraft Attitude

Chapter 21

Attitude Modeling

The attitude of satellites may be specified through two mechanisms:

- Geometrically defined orientation of the body axes relative to specified inertial directions, referred to as the Aligned and Constrained attitude specification
- A time history of attitude states in the form of quaternions or Euler angles contained in a file

However the attitude for the satellite is defined, it is used in ODTK to determine the location of antenna phase centers used in observation modeling. Antenna phase center locations are specified as fixed locations in the body frame of the satellite. The origin of the satellite body frame is at the center of mass of the satellite by default, but fixed coordinates in the body frame may also be specified for the center of mass. The inertial location of the antenna phase center is computed for use in observation modeling as follows,

$$\mathbf{R}_{ant} = \mathbf{R}_{cm} + M(\mathbf{r}_{ant} - \mathbf{r}_{cm}) \quad (21.1)$$

where \mathbf{R}_{cm} is the inertial location of the center of mass of the satellite, M is the current 3×3 rotation matrix from body fixed coordinates to inertial coordinates, \mathbf{r}_{ant} is the antenna phase center location relative to the origin of the body frame expressed in body fixed coordinates and \mathbf{r}_{cm} is the center of mass location relative to the origin of the body frame expressed in body fixed coordinates.

21.1 Antenna Phase Center Estimation

The GPS measurement models in ODTK support estimation of the antenna phase center location in the body frame of the satellite when applied to a space based GPS receiver. In this case, the antenna phase center coordinates are estimated as constants. Estimation of the antenna phase center is facilitated by applying the chain rule to the observation partials with respect to the inertial

antenna phase center location to obtain the observation partials with respect to the antenna phase center location in the body fixed frame. If α is the observed quantity, then

$$\frac{\partial \alpha}{\partial \mathbf{r}_{ant}} = \frac{\partial \alpha}{\partial \mathbf{R}_{ant}} \frac{\partial \mathbf{R}_{ant}}{\partial \mathbf{r}_{ant}} = \frac{\partial \alpha}{\partial \mathbf{R}_{ant}} M \quad (21.2)$$

Part V

State Error Transition

Chapter 22

Transitive Partial Derivatives

22.1 State Error Transition Function

The state X may contain one or more observable orbits Z^i , associated observable acceleration model parameters A^i , associated observable measurement model parameters B^i , and other observable parameters:

$$X = \begin{bmatrix} Z^1 \\ A^1 \\ B^1 \\ Z^2 \\ A^2 \\ B^2 \\ \vdots \end{bmatrix}$$

The state error transition function $\Phi(t_k, t_j)$ will be calculated as the Jacobian matrix $\partial X(t_k) / \partial X(t_j)$:

$$\Phi(t_k, t_j) = \frac{\partial X(t_k)}{\partial X(t_j)} \quad (22.1)$$

22.2 Position & Velocity Partialials

Recall Eqs. 1.24, 1.25, and 1.35.

$$Z = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \end{bmatrix}$$

for notation for spacecraft position component matrix z , velocity component matrix \dot{z} , and their composition Z . The 6x6 matrix of partial derivatives:

$$\Phi^Z(t_k, t_j) = \frac{\partial Z(t_k)}{\partial Z(t_j)}$$

moves position and velocity error $\Delta Z(t)$ from time t_j to time t_k :

$$\Delta Z(t_k) = \frac{\partial Z(t_k)}{\partial Z(t_j)} \Delta Z(t_j) = \Phi^Z(t_k, t_j) \Delta Z(t_j)$$

22.2.1 Two-Body Transition Matrix

The two-body 6x6 matrix $\Phi_{TB}^Z(t_k, t_j)$ of two-body partial derivatives for Universal Variables given by Herrick ([34] 15C)¹ has been shown to be sufficient for geocentric applications that have dense tracking data with time. But they have been shown to be *insufficient* for sparse tracking data with time. They are also likely to be insufficient for cases where two-body gravity is not the dominant force acting on the spacecraft. For these cases it is appropriate to use variational equations to incorporate perturbative effects in the orbit transition function.

22.2.2 Variational Equations Transition Matrix

Eq. 22.10 gives the general structure for the differential variational equations.

General Structure

Let $\Phi^Z(t_k, t_j)$ denote the 6x6 matrix of variational equations. It is appropriate to first derive a representation for $\Phi^Z(t_k, t_j)$. The time derivative $\dot{Z}(t)$ of the orbit $Z(t)$ is some nonlinear function $f(Z, t)$ of the orbit Z and time t :

$$\dot{Z}(t) = f(Z, t) \tag{22.2}$$

Recall Eq. 1.36:

$$Z(t) = \varphi_z \{t; Z(t_0), t_0, u(Z(\tau), \tau), t_0 \leq \tau \leq t\} \tag{22.3}$$

Differentiate Eq. 22.2 with respect to $Z(t_0)$ with t_0 fixed

¹A simple correction is required to the second equation of (Herrick 15C20B).

$$\frac{\partial \dot{Z}(t)}{\partial Z(t_0)} = \frac{\partial f(Z,t)}{\partial Z} \frac{\partial \varphi_z}{\partial Z(t_0)} = \frac{\partial \dot{Z}(t)}{\partial Z(t)} \frac{\partial Z(t)}{\partial Z(t_0)} \quad (22.4)$$

because $Z(t_0)$ is a constant. Then:

$$\frac{\partial \dot{Z}(t)}{\partial Z(t_0)} = \frac{d}{dt} \frac{\partial Z(t)}{\partial Z(t_0)} \quad (22.5)$$

Insert this into Eq. 22.4 to get:

$$\frac{d}{dt} \frac{\partial Z(t)}{\partial Z(t_0)} = \frac{\partial \dot{Z}(t)}{\partial Z(t)} \frac{\partial Z(t)}{\partial Z(t_0)} \quad (22.6)$$

Let us denote:

$$\Phi(t) = \frac{\partial Z(t)}{\partial Z(t_0)} \quad (22.7)$$

$$A(t) = \frac{\partial \dot{Z}(t)}{\partial Z(t)} \quad (22.8)$$

and note that:

$$\Phi(t) = I_{6 \times 6} \quad (22.9)$$

Eq. 22.6 has the form:

$$\dot{\Phi}(t) = A(t) \Phi(t) \quad (22.10)$$

Symbolically, write the integral to Eq. 22.10 as:

$$\Phi(t) = I_{6 \times 6} + \int_{t_0}^t A(\tau) \Phi(\tau) d\tau \quad (22.11)$$

The implementation of Eq. 22.11 benefits from simultaneous numerical integration with the numerical integration of the orbit.

The A Matrix

In essence, the problem for integration of the variational equation, Eq. 22.10, is to calculate the matrix $A(t)$ defined by Eq. 22.8:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad (22.12)$$

where:

$$A_{11} = \frac{\partial \dot{z}}{\partial z} = 0_{3 \times 3} \quad (22.13)$$

since position and velocity are instantaneously independent. Then:

$$A_{12} = \frac{\partial \dot{z}}{\partial \dot{z}} = I_{3 \times 3} \quad (22.14)$$

$$A_{21} = \frac{\partial \ddot{z}}{\partial z} \quad (22.15)$$

$$A_{22} = \frac{\partial \ddot{z}}{\partial \dot{z}} \quad (22.16)$$

Geopotential

The geopotential is a function only of position and time. Therefore $A_{22} = 0_{3 \times 3}$, and we are left with the calculation only of A_{21} .

Calculate A_{21}

Write the geopotential function U (negative potential energy per unit spacecraft mass) as:

$$\begin{aligned} U &= \frac{\mu}{r} \sum_{n=0}^{\infty} \left(\frac{a_e}{r}\right)^n \sum_{m=0}^n P_{nm}(\sin \varphi) \{C_{nm} \cos m\lambda + S_{nm} \sin m\lambda\} \\ &= \sum_{n=0}^{\infty} \sum_{m=0}^n \{C_{nm} U_{nm} + S_{nm} V_{nm}\} \end{aligned} \quad (22.17)$$

where φ and λ are geodetic latitude and longitude, $P_{nm}(\sin \varphi)$ are Legendre functions of degree n and order m , and:

$$U_{nm} = \left(\frac{\mu a_e^n}{r^{n+1}}\right) P_{nm}(\sin \varphi) \cos m\lambda \quad (22.18)$$

$$V_{nm} = \left(\frac{\mu a_e^n}{r^{n+1}}\right) P_{nm}(\sin \varphi) \sin m\lambda \quad (22.19)$$

The 3×1 gravitational acceleration matrix \ddot{z}_G^e , referred to an Earth-fixed orthonormal vector basis, is derived from the 3×1 gradient vector component matrix $\nabla^e U$ that has been applied to the geopotential function U :

$$\ddot{z}_G^e = \nabla^e U = \sum_{n=0}^{\infty} \sum_{m=0}^n \{C_{nm} \nabla^e U_{nm} + S_{nm} \nabla^e V_{nm}\} \quad (22.20)$$

The gravitational acceleration matrix \ddot{z}_G^e in Earth-fixed components is rotated to an inertial component matrix \ddot{z}_G by application of the 3×3 rotation matrix² R_{ie} :

²Precession, nutation, Greenwich rotation, and polar motion.

$$\ddot{z}_G = R_{ie} \ddot{z}_G^e = R_{ie} \sum_{n=0}^{\infty} \sum_{m=0}^n \{C_{nm} \nabla^e U_{nm} + S_{nm} \nabla^e V_{nm}\} \quad (22.21)$$

Then the A_{21} matrix has the representation:

$$A_{21} = \frac{\partial \ddot{z}_G}{\partial z} = R_{ie} \sum_{n=0}^{\infty} \sum_{m=0}^n \left\{ C_{nm} \frac{\partial (\nabla^e U_{nm})}{\partial z} + S_{nm} \frac{\partial (\nabla^e V_{nm})}{\partial z} \right\} \quad (22.22)$$

Evaluation of the second derivative terms in Eq. 22.22 is accomplished with a fast running recursive technique on indices n and m .

22.3 Air-Drag Partial

Consider:

$$X = \begin{bmatrix} Z \\ D \end{bmatrix} \quad (22.23)$$

where Z is a 6x1 substate consisting of spacecraft position and velocity, and D is a scalar 1x1 air-drag substate coefficient. The drag parameter D is modeled as an exponential Gauss-Markov sequence. Then:

$$\frac{\partial X(t_{k+1})}{\partial X(t_k)} = \begin{bmatrix} \frac{\partial X(t_{k+1})}{\partial Z(t_k)} & \frac{\partial X(t_{k+1})}{\partial D(t_k)} \end{bmatrix} = \begin{bmatrix} \frac{\partial Z(t_{k+1})}{\partial Z(t_k)} & \frac{\partial Z(t_{k+1})}{\partial D(t_k)} \\ \frac{\partial D(t_{k+1})}{\partial Z(t_k)} & \frac{\partial D(t_{k+1})}{\partial D(t_k)} \end{bmatrix} \quad (22.24)$$

where:

$$\frac{\partial D(t_{k+1})}{\partial Z(t_k)} = 0_{1 \times 6} \quad (22.25)$$

The 6x6 position and velocity Jacobian matrix $\partial Z(t_{k+1})/\partial Z(t_k)$ is calculated in the filter TimeUpdate() function using either an analytic two-body approximation, a fully perturbative numerical method, or a fast running perturbative numerical method that uses Encke's method. The 1x1 air-drag derivative $\partial D(t_{k+1})/\partial D(t_k) = \Phi(t_{k+1}, t_k)$ is given by Eq. 17.31, and the 6x1 Jacobian matrix $\partial Z(t_{k+1})/\partial D(t_k)$ is the subject of this section.

Generalize Eq. 1.36 to get:

$$X(t_{k+1}) = \varphi\{t_{k+1}; X(t_k), t_k, u(X(\tau), \tau), t_k \leq \tau \leq t_{k+1}\} \quad (22.26)$$

Insert Eq. 22.26 into the 6x1 Jacobian matrix expression $\partial X(t_{k+1})/\partial D(t_k)$ to get:

$$\frac{\partial X(t_{k+1})}{\partial D(t_k)} = \frac{\partial \varphi \{t_{k+1}; X(t_k), t_k\}}{\partial D(t_k)} = \frac{\partial \varphi \{t_{k+1}; [Z(t_k), D(t_k)]^T, t_k\}}{\partial D(t_k)} \quad (22.27)$$

The right-hand side of Eq. 22.27 can be equated to the definition of the partial derivative:

$$\frac{\partial \varphi \{t_{k+1}; [Z(t_k), D(t_k)]^T, t_k\}}{\partial D(t_k)} =$$

$$\lim_{\epsilon_D \rightarrow 0} \left\{ \frac{1}{\epsilon_D} \left\{ \varphi \{t_{k+1}; [Z(t_k), D(t_k) + \epsilon_D]^T, t_k\} - \varphi \{t_{k+1}; [Z(t_k), D(t_k)]^T, t_k\} \right\} \right\} \quad (22.28)$$

The second term on the right-hand side denotes the nominal propagation from t_k to t_{k+1} , but the first term on the right-hand side requires an additional propagation from t_k to t_{k+1} with the air-drag element modified:

$$D(t_k) \longrightarrow D(t_k) + \epsilon_D \quad (22.29)$$

Recall from the partial derivative definition that $\lim_{\epsilon_D \rightarrow 0} \{\cdot\}$ means that $\epsilon_D > 0$ is sufficiently small, and that $\epsilon_D \neq 0$. Several values for ϵ_D should be tried in order to identify the linear region for ϵ_D ; *i.e.*, an interval where $\partial X(t_{k+1})/\partial D(t_k)$ has negligible variations. The 7x1 column matrix of partial derivatives $\partial X(t_{k+1})/\partial D(t_k)$ is thus calculated according to Eq. 22.28. These propagations are executed using a fast running Encke propagator.

22.4 Solar Pressure Partial

Replace D with p in the discussion above.

Part VI

**Ground Location
Estimation**

The estimation of location in the Earth-fixed reference frame is supported by many measurement models within ODTK. To obtain the observation partials with respect to a terrestrial location in the Earth-fixed frame for estimation purposes, the chain rule is applied to these observation partials with respect to the inertial terrestrial location. If α is the observed quantity, then

$$\frac{\partial \alpha}{\partial R_{ECF}} = \frac{\partial \alpha}{\partial R_{ECI}} \frac{\partial R_{ECI}}{\partial R_{ECF}} = \frac{\partial \alpha}{\partial R_{ECI}} M \quad (22.30)$$

where M is the 3×3 ECF to ECI transformation matrix. There are two options for ground location estimation: to estimate the 3D position of the ground location or to estimate only the latitude and longitude of the ground location while the altitude remains fixed at a specified value. In the second case, where only latitude and longitude are estimated an additional step must be taken to transform the partial derivatives with respect to Cartesian coordinates to partial derivatives with respect to geodetic coordinates. Define position as the 3×1 matrix G :

$$G = \begin{bmatrix} \varphi \\ \lambda \\ H \end{bmatrix} \quad (22.31)$$

where φ is the geodetic latitude, λ is the longitude and H is the altitude above the ellipsoid. Another application of the chain rule gives

$$\frac{\partial \alpha}{\partial G} = \frac{\partial \alpha}{\partial R_{ECF}} \frac{\partial R_{ECF}}{\partial G} \quad (22.32)$$

The matrix $\frac{\partial R_{ECF}}{\partial G}$ is computed using a spherical Earth approximation as

$$\frac{\partial R_{ECF}}{\partial G} = \begin{bmatrix} \beta x_{ECF} & -y_{ECF} & x_{ECF}/R \\ \beta y_{ECF} & x_{ECF} & y_{ECF}/R \\ \theta & 0 & z_{ECF}/R \end{bmatrix} \quad (22.33)$$

where

$$\theta = \sqrt{x_{ECF}^2 + y_{ECF}^2} \quad (22.34)$$

$$\beta = \frac{-z_{ECF}}{\theta} \quad (22.35)$$

Part VII

Measurements

Chapter 23

Tracking Station Kinematics

23.1 Introduction

The purpose of this section is to derive and provide complete and consistent station kinematics results. Tracking station kinematics are used several times, and at different places, in the ODTK orbit determination program. Appropriate results include the calculation of:

- Unit range vector components on the ECEF frame from azimuth and elevation
- Azimuth and elevation from unit range vector components on the ECEF frame
- Rotation matrices for use in azimuth and elevation calculations
- Azimuth and elevation partial derivatives for use in the optimal filter
- Station position vector components on the ECEF frame, given station geodetic latitude φ , longitude λ , and height h above ellipsoid (comparison to known results enables validation)
- Station position vector components on the ECEF frame, given station geocentric latitude φ' , longitude λ , and position vector length s (comparison to known results enables validation)
- Rigorous transformations: From (φ', s) to (φ, h) and from (φ, h) to (φ', s) (comparison of transform to inverse transform – and to known results – enables validation)
- A table to show φ -dependent differences between φ' and φ for comparison to white noise RMS on angles measurements (demonstrates necessity for oblate ellipsoid in IOD angles measurement models)

Perhaps the most important result is the clarity provided in relating the various coordinate frames required for use in IOD, the simulator, and the optimal filter.

23.2 The Earth-Centered Unit Sphere

Let $[\mathbf{i}]$ denote the dextral orthonormal Earth centered inertial (ECI) vector basis, and let $[\mathbf{e}]$ denote the dextral orthonormal time-dependent Earth centered Earth fixed (ECEF) vector basis $[\mathbf{e}]$, where:

$$[\mathbf{i}] = \begin{bmatrix} \mathbf{i}_1 \\ \mathbf{i}_2 \\ \mathbf{i}_3 \end{bmatrix} \quad (23.1)$$

$$[\mathbf{e}] = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{bmatrix} \quad (23.2)$$

$$[\mathbf{e}] = R_{ei} [\mathbf{i}] \quad (23.3)$$

where \mathbf{e}_3 is coincident with the instantaneous polar axis of figure, \mathbf{e}_2 and \mathbf{e}_3 lie in the instantaneous equator of figure, \mathbf{e}_1 lies in the Greenwich meridian of figure, and $\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2$. The 3×3 rotation matrix R_{ei} is a product of four rotations: precession, nutation, Greenwich rotation, and polar motion (PNGM).

Let \mathbf{s} denote an Earth centered ground station position vector with geocentric latitude angle φ' and East longitude angle λ . Let $[\mathbf{f}]$ denote the dextral orthonormal vector basis obtained from $[\mathbf{e}]$ by an East longitude λ rotation from the Greenwich meridian to the station meridian:

$$\begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix} = \begin{bmatrix} \cos \lambda & \sin \lambda & 0 \\ -\sin \lambda & \cos \lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{bmatrix} \quad (23.4)$$

That is:

$$[\mathbf{f}] = R_{fe} [\mathbf{e}] \quad (23.5)$$

where:

$$R_{fe} = \begin{bmatrix} \cos \lambda & \sin \lambda & 0 \\ -\sin \lambda & \cos \lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (23.6)$$

Note that the ground station position vector \mathbf{s} and geocentric latitude angle φ' are contained in the meridian plane spanned by \mathbf{f}_1 and \mathbf{f}_3 . The station vector \mathbf{s} is defined by:

$$\mathbf{s} = [\mathbf{f}]^T {}_s^f \quad (23.7)$$

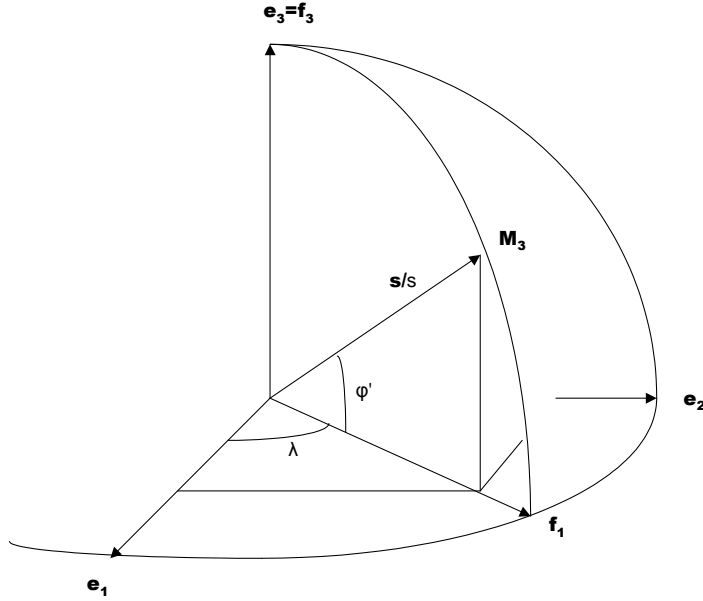


Figure 23.1: Station Longitude Rotation

where:

$$s^f = \begin{bmatrix} s \cos \varphi' \\ 0 \\ s \sin \varphi' \end{bmatrix} \quad (23.8)$$

and where:

$$s = \sqrt{\mathbf{s} \cdot \mathbf{s}} = \sqrt{(s^f)^T s^f} \quad (23.9)$$

The station unit vector $\hat{\mathbf{s}}$ is defined by:

$$\hat{\mathbf{s}} = \mathbf{s}/s = [\mathbf{f}]^T \hat{s}^f \quad (23.10)$$

where:

$$\hat{s}^f = \begin{bmatrix} \cos \varphi' \\ 0 \\ \sin \varphi' \end{bmatrix} \quad (23.11)$$

Define the dextral orthonormal geocentric station vector basis:

$$\begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \mathbf{M}_3 \end{bmatrix} = \begin{bmatrix} -\sin \varphi' & 0 & \cos \varphi' \\ 0 & -1 & 0 \\ \cos \varphi' & 0 & \sin \varphi' \end{bmatrix} \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix} \quad (23.12)$$

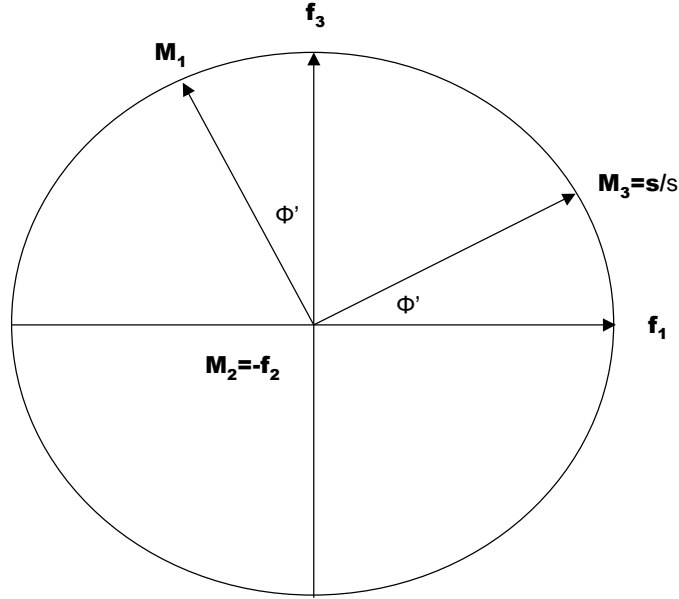


Figure 23.2: Geocentric Station Vector Basis

or:

$$[\mathbf{M}] = R_{Mf} [\mathbf{f}] \quad (23.13)$$

where:

$$R_{Mf} = \begin{bmatrix} -\sin \phi' & 0 & \cos \phi' \\ 0 & -1 & 0 \\ \cos \phi' & 0 & \sin \phi' \end{bmatrix} \quad (23.14)$$

23.3 Ellipse in Plane of \mathbf{f}_1 and \mathbf{f}_3

Let \mathbf{p} denote an Earth centered vector:

$$\mathbf{p} = [\mathbf{f}]^T p^f \quad (23.15)$$

where:

$$p^f = \begin{bmatrix} p_1^f \\ p_2^f \\ p_3^f \end{bmatrix} \quad (23.16)$$

that defines an Earth centered ellipse in the $(\mathbf{f}_1, \mathbf{f}_3)$ plane with function:

$$S(p^f) = b^2 (p_1^f)^2 + a^2 (p_3^f)^2 - a^2 b^2 = 0 \quad (23.17)$$

$$p_2^f = 0 \quad (23.18)$$

where a and b are semi-major axis and semi-minor axis respectively. Thus:

$$b^2 = a^2 (1 - e^2) \quad (23.19)$$

where e denotes eccentricity of the ellipse. Define the flattening f with:

$$f = \frac{a - b}{a} \quad (23.20)$$

Then:

$$e^2 = \frac{a^2 - b^2}{a^2} = 2f - f^2 \quad (23.21)$$

23.3.1 Normal Vector $\mathbf{n}(p^f)$

Differentiate $S(p^f)$ with respect to p^f to derive the gradient ∇_f to the ellipse:

$$\nabla_f = [\mathbf{f}]^T \begin{bmatrix} \partial S / \partial p_1^f \\ 0 \\ \partial S / \partial p_3^f \end{bmatrix} = [\mathbf{f}]^T \begin{bmatrix} 2b^2 p_1^f \\ 0 \\ 2a^2 p_3^f \end{bmatrix} \quad (23.22)$$

Define the outward normal unit vector \mathbf{n} to the ellipse:

$$\mathbf{n} = \nabla_f / \sqrt{\nabla_f \cdot \nabla_f} \quad (23.23)$$

Then:

$$\mathbf{n} = [\mathbf{f}]^T \begin{bmatrix} (1 - e^2) p_1^f / \sqrt{(1 - e^2)^2 (p_1^f)^2 + (p_3^f)^2} \\ 0 \\ p_3^f / \sqrt{(1 - e^2)^2 (p_1^f)^2 + (p_3^f)^2} \end{bmatrix} \quad (23.24)$$

23.3.2 Geodetic Latitude

Define the geodetic latitude φ with its tangent function:

$$\tan \varphi = \frac{\partial S / \partial p_3^f}{\partial S / \partial p_1^f} = \frac{p_3^f}{(1 - e^2) p_1^f} \quad (23.25)$$

Compare to Eq. 23.17 to eliminate p_3^f and solve for p_1^f :

$$p_1^f = \frac{a \cos \varphi}{\sqrt{1 - e^2 \sin^2 \varphi}} \quad (23.26)$$

Back substitute to get:

$$p_3^f = \frac{a(1-e^2)\sin\varphi}{\sqrt{1-e^2\sin^2\varphi}} \quad (23.27)$$

Define:

$$N_\varphi = \frac{a}{\sqrt{1-e^2\sin^2\varphi}} \quad (23.28)$$

Then:

$$p^f = \begin{bmatrix} p_1^f \\ p_2^f \\ p_3^f \end{bmatrix} = \begin{bmatrix} N_\varphi \cos\varphi \\ 0 \\ N_\varphi(1-e^2)\sin\varphi \end{bmatrix} \quad (23.29)$$

23.3.3 Normal Vector $\mathbf{n}(\varphi)$

Insert Eqs. 23.26 and 23.27 into Eq. 23.24 to demonstrate:

$$\mathbf{n} = [\mathbf{f}]^T n^f \quad (23.30)$$

where:

$$n^f = \begin{bmatrix} n_1^f \\ n_2^f \\ n_3^f \end{bmatrix} = \begin{bmatrix} \cos\varphi \\ 0 \\ \sin\varphi \end{bmatrix} \quad (23.31)$$

23.3.4 Station Height Above Ellipse

Let h denote the station height above the ellipse and measured along \mathbf{n} . Then:

$$\mathbf{s} = \mathbf{p} + h\mathbf{n} \quad (23.32)$$

That is:

$$\mathbf{s} = [\mathbf{f}]^T s^f \quad (23.33)$$

where:

$$s^f = \begin{bmatrix} (N_\varphi + h)\cos\varphi \\ 0 \\ ((1-e^2)N_\varphi + h)\sin\varphi \end{bmatrix} \quad (23.34)$$

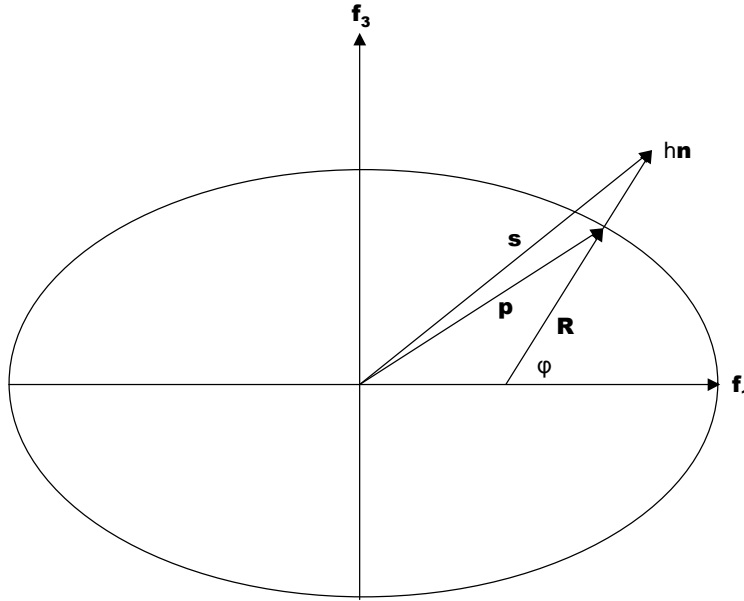


Figure 23.3: Station Position Vector

23.3.5 \mathbf{R}

Define:

$$\mathbf{R} = R\mathbf{n} \quad (23.35)$$

where:

$$R = \sqrt{\mathbf{R} \cdot \mathbf{R}} = p_3^f / \sin \varphi = N_\varphi (1 - e^2) \quad (23.36)$$

Thus:

$$\mathbf{R} = R [\mathbf{f}]^T \mathbf{n}^f = [\mathbf{f}]^T R^f \quad (23.37)$$

where:

$$R^f = \begin{bmatrix} R_1^f \\ R_2^f \\ R_3^f \end{bmatrix} = \begin{bmatrix} N_\varphi (1 - e^2) \cos \varphi \\ 0 \\ N_\varphi (1 - e^2) \sin \varphi \end{bmatrix} \quad (23.38)$$

23.3.6 Geocentric vs Geodetic Latitude

Fundamental Results

Compare Eq. 23.8 with Eq. 23.34 to write:

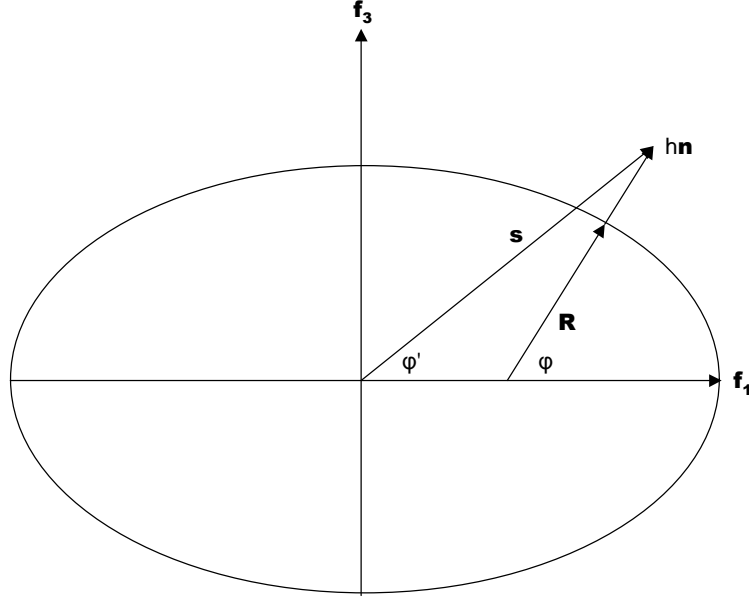


Figure 23.4: Geocentric vs Geodetic

$$\tan \varphi' = \left[\frac{(1 - e^2) N_\varphi + h}{N_\varphi + h} \right] \tan \varphi \quad (23.39)$$

where:

$$s_1^f = (N_\varphi + h) \cos \varphi = s \cos \varphi' \quad (23.40)$$

$$s_3^f = ((1 - e^2) N_\varphi + h) \sin \varphi = s \sin \varphi' \quad (23.41)$$

Given values for φ and h , then φ' can be evaluated directly using Eq. 23.39. Then using Eqs. 23.9 and 23.34:

$$s = \sqrt{(s_1^f)^2 + (s_3^f)^2} \quad (23.42)$$

Given values for φ' and s , then Eqs. 23.39 and 23.42 can be iterated for φ and h .

Newton-Raphson

Define:

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \varphi \\ h \end{bmatrix} \quad (23.43)$$

$$f(X) = \begin{bmatrix} f_1(X) \\ f_2(X) \end{bmatrix} \quad (23.44)$$

where:

$$f_1(X) = xyz - \tan \varphi' = 0 \quad (23.45)$$

$$f_2(X) = (s_1^f)^2 + (s_3^f)^2 - s^2 = 0 \quad (23.46)$$

$$x = (1 - e^2) N_\varphi + h \quad (23.47)$$

$$y = (N_\varphi + h)^{-1} \quad (23.48)$$

$$z = \tan \varphi \quad (23.49)$$

and s_1^f and s_3^f are defined by Eqs. 23.40 and 23.41.

Partial Derivatives

$$\frac{\partial f(X)}{\partial X} = \begin{bmatrix} \frac{\partial f_1}{\partial \varphi} & \frac{\partial f_1}{\partial h} \\ \frac{\partial f_2}{\partial \varphi} & \frac{\partial f_2}{\partial h} \end{bmatrix} \quad (23.50)$$

where:

$$\frac{\partial f_1}{\partial \varphi} = \frac{\partial x}{\partial \varphi} yz + x \left(\frac{\partial y}{\partial \varphi} z + y \frac{\partial z}{\partial \varphi} \right) \quad (23.51)$$

$$\frac{\partial f_1}{\partial h} = z \left(y + \frac{\partial y}{\partial h} \right) \quad (23.52)$$

$$\frac{\partial f_2}{\partial \varphi} = 2.0 \left(s_1^f \frac{\partial s_1^f}{\partial \varphi} + s_3^f \frac{\partial s_3^f}{\partial \varphi} \right) \quad (23.53)$$

$$\frac{\partial f_2}{\partial h} = 2.0 \left(s_1^f \frac{\partial s_1^f}{\partial h} + s_3^f \frac{\partial s_3^f}{\partial h} \right) \quad (23.54)$$

where:

$$\frac{\partial s_1^f}{\partial \varphi} = \frac{\partial N_\varphi}{\partial \varphi} \cos \varphi - (N_\varphi + h) \sin \varphi \quad (23.55)$$

$$\frac{\partial s_1^f}{\partial h} = \cos \varphi \quad (23.56)$$

$$\frac{\partial s_3^f}{\partial \varphi} = (1 - e^2) \frac{\partial N_\varphi}{\partial \varphi} \sin \varphi + x \cos \varphi \quad (23.57)$$

$$\frac{\partial s_3^f}{\partial h} = \sin \varphi \quad (23.58)$$

$$\frac{\partial N_\varphi}{\partial \varphi} = \frac{e^2 N_\varphi \sin \varphi \cos \varphi}{1 - e^2 \sin^2 \varphi} \quad (23.59)$$

$$\frac{\partial x}{\partial \varphi} = (1 - e^2) \frac{\partial N_\varphi}{\partial \varphi} \quad (23.60)$$

$$\frac{\partial x}{\partial h} = 1 \quad (23.61)$$

$$\frac{\partial y}{\partial \varphi} = -\frac{\partial N_\varphi}{\partial \varphi} y^2 \quad (23.62)$$

$$\frac{\partial y}{\partial h} = -y^2 \quad (23.63)$$

$$\frac{\partial z}{\partial \varphi} = \sec^2 \varphi \quad (23.64)$$

$$\frac{\partial z}{\partial h} = 0 \quad (23.65)$$

Algorithm

$$X^{(0)} = \begin{bmatrix} \varphi' \\ 0 \end{bmatrix} \quad (23.66)$$

$$X^{(k+1)} = X^{(k)} - \Delta X^{(k)}, \quad k \in \{0, 1, \dots, n\} \quad (23.67)$$

$$\Delta X^{(k)} = \left[\frac{\partial f(X)}{\partial X} \right]_{(k)}^{-1} [f(X)]_{(k)} \quad (23.68)$$

$$|\Delta X^{(n)}| < \epsilon \quad (23.69)$$

23.3.7 Topocentric Geodetic Vector Basis [N]

Define station zenith with \mathbf{N}_3 :

$$\mathbf{N}_3 = \mathbf{n} = [\mathbf{f}]^T \begin{bmatrix} \cos \varphi \\ 0 \\ \sin \varphi \end{bmatrix} \quad (23.70)$$

Obtain a dextral orthonormal triad [N] with \mathbf{N}_1 as origin for station North reference:

$$\tilde{\mathbf{N}}_2 = \mathbf{n} \times \mathbf{f}_3 \quad (23.71)$$

$$\mathbf{N}_2 = \tilde{\mathbf{N}}_2 / \sqrt{\tilde{\mathbf{N}}_2 \cdot \tilde{\mathbf{N}}_2} = [\mathbf{f}]^T \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \quad (23.72)$$

$$\mathbf{N}_1 = \mathbf{N}_2 \times \mathbf{N}_3 = [\mathbf{f}]^T \begin{bmatrix} -\sin \varphi \\ 0 \\ \cos \varphi \end{bmatrix} \quad (23.73)$$

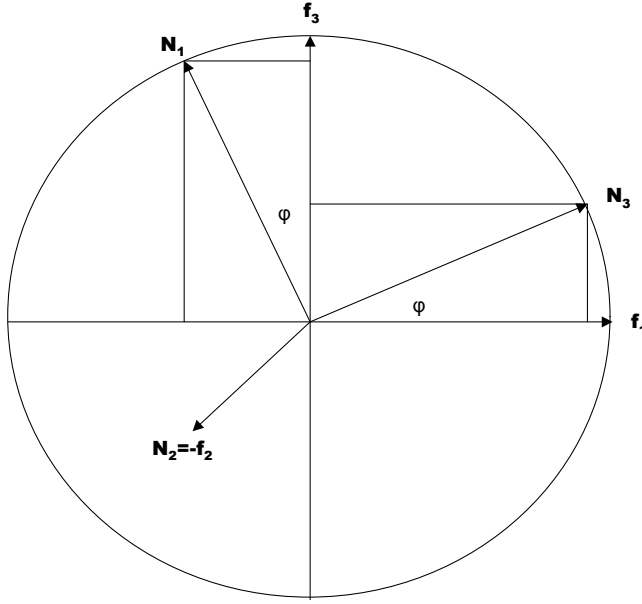


Figure 23.5: Vector Basis for Angles Measurements

In matrix form we have:

$$\begin{bmatrix} \mathbf{N}_1 \\ \mathbf{N}_2 \\ \mathbf{N}_3 \end{bmatrix} = \begin{bmatrix} -\sin \varphi & 0 & \cos \varphi \\ 0 & -1 & 0 \\ \cos \varphi & 0 & \sin \varphi \end{bmatrix} \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix} \quad (23.74)$$

That is:

$$[\mathbf{N}] = R_{Nf} [\mathbf{f}] \quad (23.75)$$

where:

$$R_{Nf} = \begin{bmatrix} -\sin \varphi & 0 & \cos \varphi \\ 0 & -1 & 0 \\ \cos \varphi & 0 & \sin \varphi \end{bmatrix} \quad (23.76)$$

This particular definition for rotation matrix R_{Nf} is necessary in order to simultaneously: Put \mathbf{N}_3 at station zenith, define station North origin with \mathbf{N}_1 , and obtain a *dextral* orthonormal triad $[\mathbf{N}]$.

23.3.8 Topocentric Geocentric Vector Basis $[\mathbf{M}]$

The rotation between geocentric latitude φ' and geodetic latitude φ is represented by a rotation from vector basis $[\mathbf{N}]$ to vector basis $[\mathbf{M}]$ by angle Δ where:

$$\Delta = \varphi - \varphi' \quad (23.77)$$

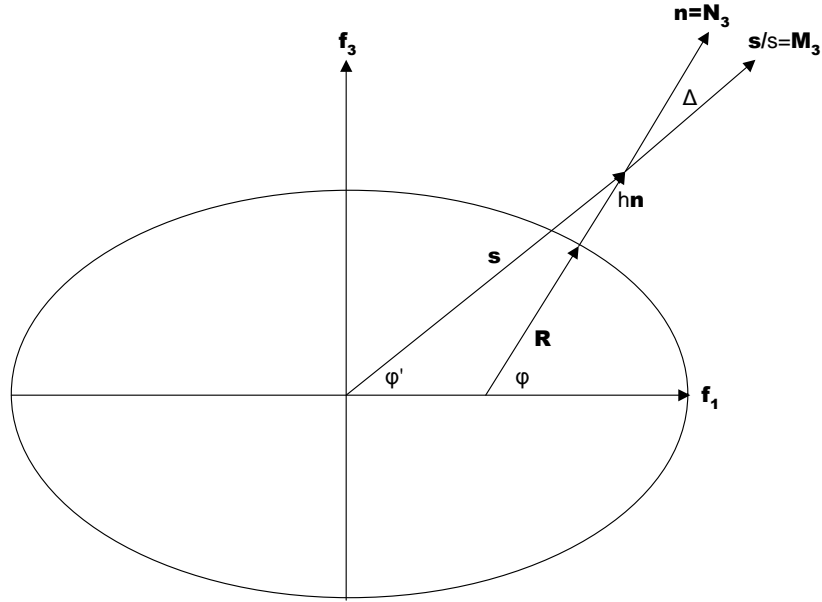


Figure 23.6: Latitude Rotation

Define:

$$\begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \\ \mathbf{M}_3 \end{bmatrix} = \begin{bmatrix} \cos \Delta & 0 & \sin \Delta \\ 0 & 1 & 0 \\ -\sin \Delta & 0 & \cos \Delta \end{bmatrix} \begin{bmatrix} \mathbf{N}_1 \\ \mathbf{N}_2 \\ \mathbf{N}_3 \end{bmatrix} \quad (23.78)$$

That is:

$$[\mathbf{M}] = R_{MN} [\mathbf{N}] \quad (23.79)$$

where:

$$R_{MN} = \begin{bmatrix} \cos \Delta & 0 & \sin \Delta \\ 0 & 1 & 0 \\ -\sin \Delta & 0 & \cos \Delta \end{bmatrix} \quad (23.80)$$

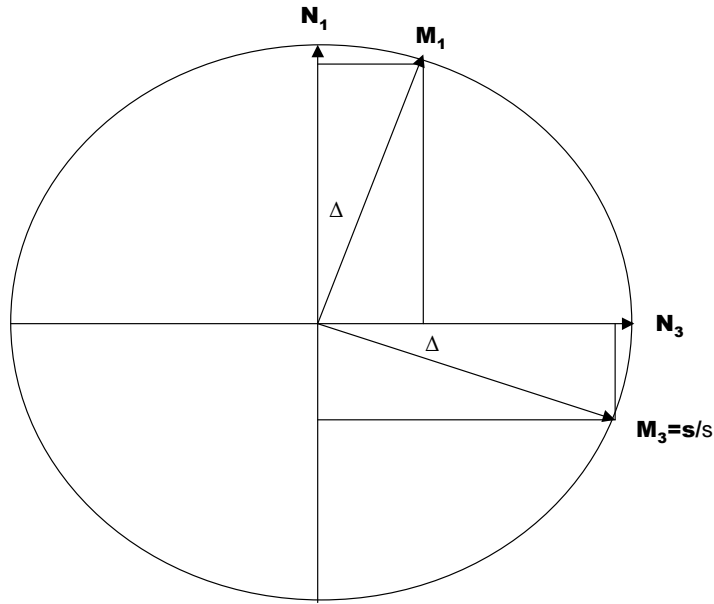


Figure 23.7: Latitude Difference Rotation

Check

Compare Eq. 23.79 with Eq. 23.13:

$$[\mathbf{N}] = R_{Nf} [\mathbf{f}] \quad (23.81)$$

where:

$$R_{NM} = R_{MN}^T \quad (23.82)$$

and:

$$R_{Nf} = R_{NM} R_{Mf} \quad (23.83)$$

to get:

$$R_{Nf} = \begin{bmatrix} -\sin(\varphi' + \Delta) & 0 & \cos(\varphi' + \Delta) \\ 0 & -1 & 0 \\ \cos(\varphi' + \Delta) & 0 & \sin(\varphi' + \Delta) \end{bmatrix} \quad (23.84)$$

Compare Eq. 23.84 with Eq. 23.76 to confirm Eq. 23.77: $\Delta = \varphi - \varphi'$.

23.4 Earth Fixed Vector Basis [e]

23.4.1 From [f] to [e]

Eq. 23.5 presents:

$$[\mathbf{f}] = R_{fe} [\mathbf{e}] \quad (23.85)$$

where:

$$R_{fe} = \begin{bmatrix} \cos \lambda & \sin \lambda & 0 \\ -\sin \lambda & \cos \lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (23.86)$$

Run the matrix transpose operator through Eq. 23.85 to get:

$$[\mathbf{f}]^T = [\mathbf{e}]^T R_{fe}^T = [\mathbf{e}]^T R_{ef} \quad (23.87)$$

where:

$$R_{fe}^T = R_{ef} = \begin{bmatrix} \cos \lambda & -\sin \lambda & 0 \\ \sin \lambda & \cos \lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (23.88)$$

23.4.2 The Ellipsoid

Insert Eq. 23.87 into Eq. 23.15 to get:

$$\mathbf{p} = [\mathbf{e}]^T R_{ef} p^f = [\mathbf{e}]^T p^e \quad (23.89)$$

where:

$$p^e = R_{ef} p^f \quad (23.90)$$

Insert Eqs. 23.88 and 23.29 into Eq. 23.90 to get:

$$p^e = \begin{bmatrix} N_\varphi \cos \varphi \cos \lambda \\ N_\varphi \cos \varphi \sin \lambda \\ N_\varphi (1 - e^2) \sin \varphi \end{bmatrix} \quad (23.91)$$

where $p^e \equiv p^e(\varphi, \lambda)$ specifies each point on the Earth ellipsoid.

23.4.3 Station Vector

Insert Eqs. 23.87 into Eq. 23.33:

$$\mathbf{s} = [\mathbf{e}]^T R_{ef} s^f = [\mathbf{e}]^T s^e \quad (23.92)$$

That is:

$$s^e = R_{ef} s^f \quad (23.93)$$

Geodetic: Given h and φ

Insert Eq. 23.34 into Eq. 23.93:

$$s^e = \begin{bmatrix} (N_\varphi + h) \cos \varphi \cos \lambda \\ (N_\varphi + h) \cos \varphi \sin \lambda \\ ((1 - e^2) N_\varphi + h) \sin \varphi \end{bmatrix} \quad (23.94)$$

Geocentric: Given s and φ'

Insert Eq. 23.8 into Eq. 23.93:

$$s^e = \begin{bmatrix} s \cos \varphi' \cos \lambda \\ s \cos \varphi' \sin \lambda \\ s \sin \varphi' \end{bmatrix} \quad (23.95)$$

23.4.4 Normal Vector $\mathbf{n}(\varphi)$

Insert Eq. 23.87 into Eq. 23.30 to get:

$$\mathbf{n} = [\mathbf{e}]^T R_{ef} n^f = [\mathbf{e}]^T n^e \quad (23.96)$$

where:

$$n^e = R_{ef} n^f = \begin{bmatrix} \cos \varphi \cos \lambda \\ \cos \varphi \sin \lambda \\ \sin \varphi \end{bmatrix} \quad (23.97)$$

23.4.5 R

Insert Eq. 23.87 into Eq. 23.37:

$$\mathbf{R} = R [\mathbf{f}]^T n^f = [\mathbf{e}]^T R_{ef} R^f = [\mathbf{e}]^T R^e \quad (23.98)$$

where:

$$R^e = R_{ef} R^f \quad (23.99)$$

Thus:

$$R^e = \begin{bmatrix} R_1^e \\ R_2^e \\ R_3^e \end{bmatrix} = \begin{bmatrix} N_\varphi (1 - e^2) \cos \varphi \cos \lambda \\ N_\varphi (1 - e^2) \cos \varphi \sin \lambda \\ N_\varphi (1 - e^2) \sin \varphi \end{bmatrix} \quad (23.100)$$

Chapter 24

Angles

24.1 Unit Range Vector

The instantaneous ground station unit range vector \mathbf{L} is defined by:

$$\mathbf{L} = \boldsymbol{\rho} / \rho \quad (24.1)$$

where:

$$\boldsymbol{\rho} = \mathbf{r} - \mathbf{s} \quad (24.2)$$

$$\rho = \sqrt{\boldsymbol{\rho} \cdot \boldsymbol{\rho}} \quad (24.3)$$

and where \mathbf{r} is the Earth centered spacecraft position vector. The 3×1 matrix L^i of inertial components for \mathbf{L} is defined by:

$$\mathbf{L} = [\mathbf{i}]^T L^i \quad (24.4)$$

where:

$$L^i = \begin{bmatrix} L_1^i \\ L_2^i \\ L_3^i \end{bmatrix} \quad (24.5)$$

24.2 Azimuth and Elevation

Let us denote azimuth and elevation with A and E , and define:

$$\beta = 2\pi - A \quad (24.6)$$

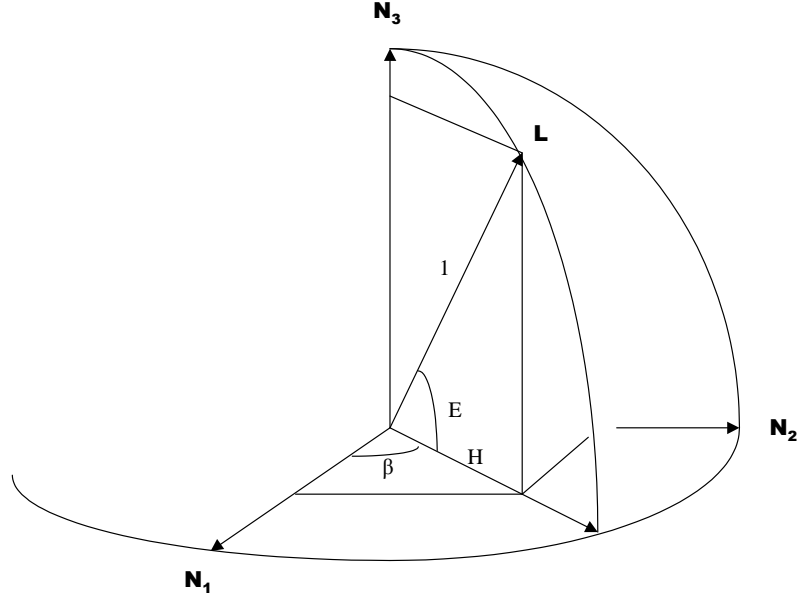


Figure 24.1: Azimuth-Elevation Description

24.2.1 Construct \mathbf{L} from Azimuth and Elevation

\mathbf{L} is constructed here from an azimuth-elevation pair.

$$\mathbf{L} = [\mathbf{N}]^T L^N \quad (24.7)$$

where:

$$L^N = \begin{bmatrix} L_1^N \\ L_2^N \\ L_3^N \end{bmatrix} = \begin{bmatrix} \cos E \cos A \\ -\cos E \sin A \\ \sin E \end{bmatrix} \quad (24.8)$$

The following notational convention is used repeatedly in that which follows: Given any orthonormal rotation matrix $R_{\alpha\beta}$, then $R_{\beta\alpha} = R_{\alpha\beta}^T$. Given the value for φ , insert Eq. 23.75 into Eq. 24.7 to get:

$$\mathbf{L} = [\mathbf{f}]^T R_{fN} L^N = [\mathbf{f}]^T L^f \quad (24.9)$$

where:

$$L^f = R_{fN} L^N \quad (24.10)$$

Into Eq. 24.9 insert Eq. 23.5 to get:

$$\mathbf{L} = [\mathbf{e}]^T R_{ef} R_{fN} L^N = [\mathbf{e}]^T L^e \quad (24.11)$$

where:

$$L^e = R_{ef}R_{fN}L^N \quad (24.12)$$

If L^i is desired, insert Eq. 23.3 into Eq. 24.11:

$$\mathbf{L} = [\mathbf{i}]^T R_{ie}R_{ef}R_{fN}L^N = [\mathbf{i}]^T L^i \quad (24.13)$$

to get:

$$L^i = R_{ie}R_{ef}R_{fN}L^N \quad (24.14)$$

24.2.2 Construct Azimuth and Elevation from \mathbf{L}

Given L^i , construct an azimuth-elevation pair.

$$\mathbf{L} = [\mathbf{i}]^T L^i = [\mathbf{i}]^T R_{ie}R_{ef}R_{fN}L^N \quad (24.15)$$

so:

$$L^N = R_{Nf}R_{fe}R_{ei}L^i \quad (24.16)$$

Solve Eq. 24.8 for A and E :

$$\begin{bmatrix} A \\ E \end{bmatrix} = \begin{bmatrix} \tan^{-1} \left(-L_2^N / L_1^N \right)_{4QUADRANT} \\ \tan^{-1} \left(L_3^N / \sqrt{(L_1^N)^2 + (L_2^N)^2} \right) \end{bmatrix} \quad (24.17)$$

24.2.3 Partial Derivatives of Azimuth and Elevation

Azimuth and Elevation with respect to L^N

From Eq. 24.8:

$$\sin A = -L_2^N / \cos E \quad (24.18)$$

$$\cos A = L_1^N / \cos E \quad (24.19)$$

$$\sin E = L_3^N \quad (24.20)$$

$$\cos E = \sqrt{(L_1^N)^2 + (L_2^N)^2} \quad (24.21)$$

$$\tan A = -L_2^N (L_1^N)^{-1} \quad (24.22)$$

$$\tan E = L_3^N / \cos E \quad (24.23)$$

Differentiate Eqs. 24.22 and 24.23 to get the tensor covariant derivatives:

$$\frac{\partial A}{\partial L^N} = \left[\begin{array}{ccc} \frac{\partial A}{\partial L_1^N} & \frac{\partial A}{\partial L_2^N} & \frac{\partial A}{\partial L_3^N} \end{array} \right] = \left[\begin{array}{ccc} -\sin A / \cos E & -\cos A / \cos E & 0 \end{array} \right] \quad (24.24)$$

$$\frac{\partial E}{\partial L^N} = \left[\begin{array}{ccc} \frac{\partial E}{\partial L_1^N} & \frac{\partial E}{\partial L_2^N} & \frac{\partial E}{\partial L_3^N} \end{array} \right] = \left[\begin{array}{ccc} -\sin E \cos A & \sin E \sin A & \cos E \end{array} \right] \quad (24.25)$$

Azimuth and Elevation with respect to L^e

From:

$$L^N = R_{Ne} L^e \quad (24.26)$$

derive:

$$\frac{\partial E}{\partial L^e} = \frac{\partial E}{\partial L^N} \frac{\partial L^N}{\partial L^e} = \frac{\partial E}{\partial L^N} R_{Ne} \quad (24.27)$$

$$\frac{\partial A}{\partial L^e} = \frac{\partial A}{\partial L^N} \frac{\partial L^N}{\partial L^e} = \frac{\partial A}{\partial L^N} R_{Ne} \quad (24.28)$$

Azimuth and Elevation with respect to L^i

From

$$L^e = R_{ei} L^i \quad (24.29)$$

derive:

$$\frac{\partial E}{\partial L^i} = \frac{\partial E}{\partial L^e} \frac{\partial L^e}{\partial L^i} = \frac{\partial E}{\partial L^e} R_{ei} \quad (24.30)$$

$$\frac{\partial A}{\partial L^i} = \frac{\partial A}{\partial L^e} \frac{\partial L^e}{\partial L^i} = \frac{\partial A}{\partial L^e} R_{ei} \quad (24.31)$$

Azimuth and Elevation with respect to x

Let x denote the 3×1 matrix of inertial components for the spacecraft position vector. Then:

$$\frac{\partial E}{\partial x} = \frac{\partial E}{\partial L^i} \frac{\partial L^i}{\partial x} \quad (24.32)$$

$$\frac{\partial A}{\partial x} = \frac{\partial A}{\partial L^i} \frac{\partial L^i}{\partial x} \quad (24.33)$$

It remains to find $\partial L^i / \partial x$. From:

$$\boldsymbol{\rho} = \rho \mathbf{L} = \mathbf{r} - \mathbf{s} \quad (24.34)$$

derive:

$$\mathbf{L} = (\mathbf{r} - \mathbf{s}) \rho^{-1} \quad (24.35)$$

Take the inner product of \mathbf{L} with Eq. 24.35 to get:

$$\rho = \mathbf{L} \cdot (\mathbf{r} - \mathbf{s}) \quad (24.36)$$

Convert the vectors in the last two equations to component arrays on the inertial frame:

$$L^i = (x - s^i) \rho^{-1} \quad (24.37)$$

$$\rho = (L^i)^T (x - s^i) \quad (24.38)$$

Insert Eq. 24.38 into Eq. 24.37 to get:

$$L^i = (x - s^i) \left((L^i)^T (x - s^i) \right)^{-1} \quad (24.39)$$

Differentiate with respect to x to get:

$$\left[I_{3 \times 3} + (x - s^i) (x - s^i)^T / \rho^2 \right] \frac{\partial L^i}{\partial x} = \left[(I_{3 \times 3} / \rho) - (x - s^i) (L^i)^T / \rho^2 \right] \quad (24.40)$$

$$\left[\rho^2 I_{3 \times 3} + (x - s^i) (x - s^i)^T \right] \frac{\partial L^i}{\partial x} = \left[(\rho I_{3 \times 3}) - (x - s^i) (L^i)^T \right] \quad (24.41)$$

Multiply through by the inverse of the first factor to get the desired result:

$$\frac{\partial L^i}{\partial x} = \left[\rho^2 I_{3 \times 3} + (x - s^i) (x - s^i)^T \right]^{-1} \left[(\rho I_{3 \times 3}) - (x - s^i) (L^i)^T \right] \quad (24.42)$$

24.2.4 Construct L^e from Azimuth and Elevation

Form L^N with Eq. 24.8. Then from Eq. 24.12:

$$L^e = R_{ef} R_{fN} L^N = R_{eN} L^N \quad (24.43)$$

where:

$$R_{eN} = \begin{bmatrix} -\sin \varphi \cos \lambda & \sin \lambda & \cos \varphi \cos \lambda \\ -\sin \varphi \sin \lambda & -\cos \lambda & \cos \varphi \sin \lambda \\ \cos \varphi & 0 & \sin \varphi \end{bmatrix} \quad (24.44)$$

24.2.5 Construct Azimuth and Elevation from L^e

From Eq. 24.43:

$$L^N = R_{Ne} L^e \quad (24.45)$$

where, using Eq. 24.44:

$$R_{Ne} = R_{eN}^T \quad (24.46)$$

From Eq. 24.17:

$$\begin{bmatrix} A \\ E \end{bmatrix} = \begin{bmatrix} \tan^{-1}(-L_2^N/L_1^N)_{4QUADRANT} \\ \tan^{-1}\left(L_3^N/\sqrt{(L_1^N)^2 + (L_2^N)^2}\right) \end{bmatrix} \quad (24.47)$$

24.3 Direction Cosines

ODTK supports the processing of *direction cosine* observations. These observations are referenced to a local dextral horizontal reference frame, the baseline directions of which may rotated relative to the frame designated by N (North, West, Zenith) in the description of Azimuth and Elevation. The direction-cosine triplet is defined by the components of \mathbf{L} referred to the topocentric basis \mathbf{F} . Analogous to our presentation of the topocentric basis for azimuth and elevation angles:

$$\mathbf{L} = [\mathbf{F}]^T L^F \quad (24.48)$$

where:

$$L^F = \begin{bmatrix} L_1^F \\ L_2^F \\ L_3^F \end{bmatrix} \quad (24.49)$$

The \mathbf{F} basis is rotated about the \mathbf{F}_3 (or \mathbf{N}_3) axis by the amount $-\Delta\beta$ in the right-handed sense

$$R_{FN} = \begin{bmatrix} -\cos(\Delta\beta) & -\sin(\Delta\beta) & 0 \\ \sin(\Delta\beta) & -\cos(\Delta\beta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (24.50)$$

$$L^F = R_{FN} L^N \quad (24.51)$$

ODTK supports the processing of L_1^F and L_2^F as measurement types that originate from the Air Force Space Surveillance System (AFSSS) "Fence" (formerly known as the Naval Space Surveillance System, or NAVSPASUR). Only two direction-cosine observations are needed in practice, since the third component is mathematically related to the other two

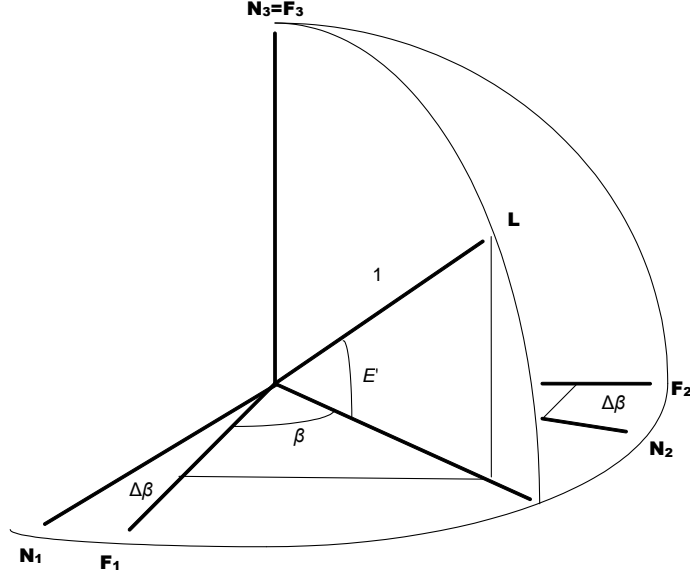


Figure 24.2: Direction Cosines Description

$$L_3^F = \sqrt{(L_1^F)^2 + (L_2^F)^2}. \quad (24.52)$$

Given the value for φ , combine Eq. 23.75, Eq. 23.73 and Eq. 24.51 to get:

$$\mathbf{L} = [\mathbf{F}]^T R_{FN} L^N = [\mathbf{F}]^T L^F \quad (24.53)$$

Into Eq. 24.53 insert Eq. 23.5 to get:

$$\mathbf{L} = [\mathbf{e}]^T R_{ef} R_{fN} R_{NF} L^F = [\mathbf{e}]^T L^e \quad (24.54)$$

where:

$$L^e = R_{ef} R_{fN} R_{NF} L^F \quad (24.55)$$

If L^i is desired, insert Eq. 23.3 into Eq. 24.54:

$$\mathbf{L} = [\mathbf{i}]^T R_{ie} R_{ef} R_{fN} R_{NF} L^F = [\mathbf{i}]^T L^i \quad (24.56)$$

to get:

$$L^i = R_{ie} R_{ef} R_{fN} R_{NF} L^F \quad (24.57)$$

24.3.1 Construct Direction Cosines L_1^F and L_2^F from \mathbf{L}

Given L^i , construct a direction-cosine set given

$$\mathbf{L} = [\mathbf{i}]^T L^i = [\mathbf{i}]^T R_{ie} R_{ef} R_{fN} R_{NF} L^F \quad (24.58)$$

such that:

$$L^F = R_{FN} R_{Nf} R_{fe} R_{ei} L^i \quad (24.59)$$

24.3.2 Partial Derivatives of Direction Cosines L_1^F and L_2^F with respect to L^N

From Eq. 24.51:

$$L_1^F = -\cos(\Delta\beta)L_1^N - \sin(\Delta\beta)L_2^N \quad (24.60)$$

$$L_2^F = \sin(\Delta\beta)L_1^N - \cos(\Delta\beta)L_2^N \quad (24.61)$$

$$L_3^F = L_3^N = \sqrt{(L_1^N)^2 + (L_2^N)^2} \quad (24.62)$$

Differentiate Eqs. 24.60 and 24.61 to get the tensor-covariant derivatives:

$$\frac{\partial L_1^F}{\partial L^N} = \left[\begin{array}{ccc} \frac{\partial L_1^F}{\partial L_1^N} & \frac{\partial L_1^F}{\partial L_2^N} & \frac{\partial L_1^F}{\partial L_3^N} \end{array} \right] = \left[\begin{array}{ccc} -\cos(\Delta\beta) & -\sin(\Delta\beta) & 0 \end{array} \right] \quad (24.63)$$

$$\frac{\partial L_2^F}{\partial L^N} = \left[\begin{array}{ccc} \frac{\partial L_2^F}{\partial L_1^N} & \frac{\partial L_2^F}{\partial L_2^N} & \frac{\partial L_2^F}{\partial L_3^N} \end{array} \right] = \left[\begin{array}{ccc} \sin(\Delta\beta) & -\cos(\Delta\beta) & 0 \end{array} \right] \quad (24.64)$$

L_1^F and L_2^F with respect to L^e

From:

$$L^F = R_{Fe} L^e \quad (24.65)$$

derive:

$$\frac{\partial L_1^F}{\partial L^e} = \frac{\partial L_1^F}{\partial L^N} \frac{\partial L^N}{\partial L^e} = \frac{\partial L_1^F}{\partial L^N} R_{Ne} \quad (24.66)$$

$$\frac{\partial L_2^F}{\partial L^e} = \frac{\partial L_2^F}{\partial L^N} \frac{\partial L^N}{\partial L^e} = \frac{\partial L_2^F}{\partial L^N} R_{Ne} \quad (24.67)$$

L_1^F and L_2^F with respect to L^i

From Eq. 24.65, derive:

$$\frac{\partial L_1^F}{\partial L^i} = \frac{\partial L_1^F}{\partial L^e} \frac{\partial L^e}{\partial L^i} = \frac{\partial L_1^F}{\partial L^e} R_{ei} \quad (24.68)$$

$$\frac{\partial L_2^F}{\partial L^i} = \frac{\partial L_2^F}{\partial L^e} \frac{\partial L^e}{\partial L^i} = \frac{\partial L_2^F}{\partial L^e} R_{ei} \quad (24.69)$$

L_1^F and L_2^F with respect to x

Let x denote the 3×1 matrix of inertial components for the spacecraft position vector. Then:

$$\frac{\partial L_1^F}{\partial x} = \frac{\partial L_1^F}{\partial L^i} \frac{\partial L^i}{\partial x} \quad (24.70)$$

$$\frac{\partial L_2^F}{\partial x} = \frac{\partial L_2^F}{\partial L^i} \frac{\partial L^i}{\partial x} \quad (24.71)$$

where $\partial L^i / \partial x$ is given by Eq. 24.42.

24.4 X/Y Angles

ODTK supports three types of X/Y angles, where the distinction between the types can be reduced to the unique definitions of the local reference frame in which the angles are measured. After a transformation to the correct local reference frame, the X/Y angle values and their partial derivatives are computed by a single set of formulas. We will describe the local reference frames via their relationship with the conventional topocentric frame (South, East, Zenith) which is related to the frame designated by N (North, West, Zenith) in the description of Azimuth and Elevation as

$$R_{TN} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (24.72)$$

$$\rho^T = R_{TN} \rho^N \quad (24.73)$$

24.4.1 X/Y East-West Reference Frame

The local reference frame for the computation of X/Y angles of the East-West type has basis vectors in the local East, North and Zenith directions. The local frame is related to the conventional topocentric frame as

$$R_{XT} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (24.74)$$

$$\rho^X = R_{XT}\rho^T \quad (24.75)$$

24.4.2 X/Y North-South Reference Frame

The local reference frame for the computation of X/Y angles of the North-South type has basis vectors in the local North, West and Zenith directions. The local frame is related to the conventional topocentric frame (South, East, Zenith) as

$$R_{XT} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (24.76)$$

$$\rho^X = R_{XT}\rho^T \quad (24.77)$$

24.4.3 X/Y Z Reference Frame

The local reference frame for the computation of X/Y angles with an accompanying Z angle is related to the conventional topocentric frame (South, East, Zenith) as

$$R_{XT} = \begin{bmatrix} -\cos(Z) & -\sin(Z) & 0 \\ \sin(Z) & -\cos(Z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (24.78)$$

$$\rho^X = R_{XT}\rho^T \quad (24.79)$$

The Z angle specifies a rotation about the Zenith direction in a right handed sense to specify the direction of the X axis where a rotation angle of zero corresponds to the X axis oriented with local North. X/Y angles of the North-South type are equivalent to X/Y Z angles with a Z angle of zero degrees. X/Y angles of the East-West type are equivalent to X/Y Z angles with a Z angle of 270 degrees.

24.4.4 X and Y Angles

The X angle is computed in the local frame as the angle from the Zenith direction to the projection of the relative position vector into the Y - Z plane. The X angle is defined to be positive for a rotation in a right handed sense about the X axis of the local frame.

$$X = \tan^{-1} \left[\frac{-\rho_2^X}{\rho_3^X} \right] \quad (24.80)$$

The Y angle is measured in the local frame as the angle from the relative position vector to the projection of the relative position vector into the Y - Z plane. The Y angle is defined to be positive for a rotation in a right handed sense about the Y axis of the local frame.

$$Y = \tan^{-1} \left[\frac{\rho_1^X}{\sqrt{(\rho_2^X)^2 + (\rho_3^X)^2}} \right] \quad (24.81)$$

24.4.5 Partial Derivatives of X and Y Angles

The partial derivative of the X angle with respect to components of the relative position vector in the local frame is given by

$$\frac{\partial X}{\partial \rho^X} = \left[0 \quad \frac{-\rho_3^X}{(\rho_2^X)^2 + (\rho_3^X)^2} \quad \frac{-\rho_2^X}{(\rho_2^X)^2 + (\rho_3^X)^2} \right] \quad (24.82)$$

The partial derivative of the Y angle with respect to components of the relative position vector in the local frame is given by

$$\frac{\partial Y}{\partial \rho^X} = \left[\frac{\sqrt{(\rho_2^X)^2 + (\rho_3^X)^2}}{\rho^2} \quad \frac{-\rho_1^X \rho_2^X}{\rho^2 \sqrt{(\rho_2^X)^2 + (\rho_3^X)^2}} \quad \frac{-\rho_1^X \rho_3^X}{\rho^2 \sqrt{(\rho_2^X)^2 + (\rho_3^X)^2}} \right] \quad (24.83)$$

24.5 Ground Based Tracker

Let us denote the right ascension and declination of \mathbf{L} with $\eta_1 = \alpha$ and $\eta_2 = \delta$, referred to the inertial orthonormal vector basis $[\mathbf{i}]$. Recall Eqs. 24.4 and 24.5 to define α and δ with:

$$L^i = \begin{bmatrix} L_1^i \\ L_2^i \\ L_3^i \end{bmatrix} = \begin{bmatrix} \cos \delta \cos \alpha \\ \cos \delta \sin \alpha \\ \sin \delta \end{bmatrix} \quad (24.84)$$

from which:

$$\eta = \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} = \begin{bmatrix} \alpha \\ \delta \end{bmatrix} = \begin{bmatrix} \tan^{-1} \left((L_2^i / L_1^i)_{4\text{-quadrant}} \right) \\ \tan^{-1} \left(L_3^i / \sqrt{(L_1^i)^2 + (L_2^i)^2} \right) \end{bmatrix} \quad (24.85)$$

24.5.1 Partials

We need a method to calculate the 2×3 matrix of partials $\partial \eta / \partial z$. Construct the chain:

$$\frac{\partial \eta}{\partial z} = \frac{\partial \eta}{\partial L^i} \frac{\partial L^i}{\partial z} \quad (24.86)$$

Let us denote the 3×1 matrix of ECI ground station position components with s^i . Then recall Eqs. 1.24, 24.2, and 24.3 and to write:

$$\rho^2 = z^T z + (s^i)^T (s^i) - 2 (s^i)^T z \quad (24.87)$$

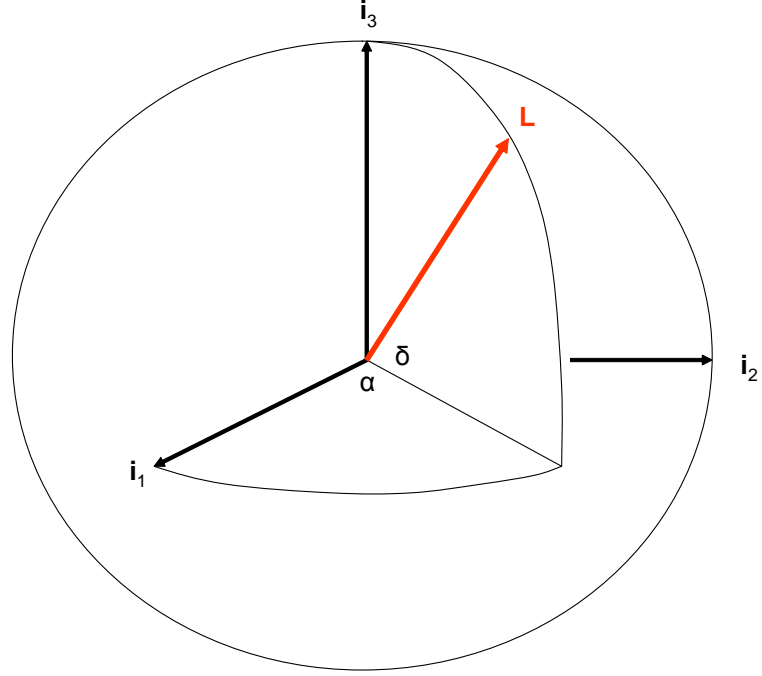


Figure 24.3: Unit Range Vector Right Ascension & Declination

$$\rho L^i = z - s^i \quad (24.88)$$

Differentiate the latter two equations with respect to z to get:

$$\frac{\partial \rho}{\partial z} = \frac{1}{\rho} (z - s^i)^T \quad (24.89)$$

$$\frac{\partial L^i}{\partial z} = \frac{1}{\rho} \left[I_{3 \times 3} - \frac{\partial \rho}{\partial z} L^i \right] \quad (24.90)$$

Thus:

$$\frac{\partial L^i}{\partial z} = \frac{1}{\rho} \left[I_{3 \times 3} - \frac{1}{\rho} (z - s^i)^T L^i \right] \quad (24.91)$$

Use and differentiate Eq. 24.84 with respect to L^i :

$$\frac{\partial \eta}{\partial L^i} = \begin{bmatrix} \frac{\partial \alpha}{\partial L_1^i} & \frac{\partial \alpha}{\partial L_2^i} & \frac{\partial \alpha}{\partial L_3^i} \\ \frac{\partial \delta}{\partial L_1^i} & \frac{\partial \delta}{\partial L_2^i} & \frac{\partial \delta}{\partial L_3^i} \end{bmatrix} \quad (24.92)$$

where:

$$\frac{\partial \alpha}{\partial L_1^i} = -\frac{L_2}{L_1^2 + L_2^2} \quad (24.93)$$

$$\frac{\partial \alpha}{\partial L_2^i} = \frac{L_1}{L_1^2 + L_2^2} \quad (24.94)$$

$$\frac{\partial \alpha}{\partial L_3^i} = 0 \quad (24.95)$$

$$\frac{\partial \delta}{\partial L_1^i} = -\frac{L_1 L_3}{\sqrt{L_1^2 + L_2^2}} \quad (24.96)$$

$$\frac{\partial \delta}{\partial L_2^i} = -\frac{L_2 L_3}{\sqrt{L_1^2 + L_2^2}} \quad (24.97)$$

$$\frac{\partial \delta}{\partial L_3^i} = \sqrt{L_1^2 + L_2^2} \quad (24.98)$$

The desired measurement partials are assembled with Eq. 24.86.

24.6 Space Based Tracker

Repeat the section above where vector \mathbf{s} denotes space station position (not ground station position).

24.7 Earth Ellipsoid Values

24.7.1 NIMA/NASA EGM96

(<http://www.nima.mil/GandG/wgs-84/egm96.html>)

a	6378137.0m
f	1.0/298.2572235630
μ	$0.3986004418 \times 10^{15} m^3/s^2$
ω_E	$7292115.0 \times 10^{-11} rad/s$

Derived Values

f	$3.3528106647474807 \times 10^{-3}$
f^2	$1.124133935364444355382 \times 10^{-5}$
e^2	$6.694379990141316996137 \times 10^{-3}$
e	$8.181919084262149433480 \times 10^{-2}$
b^2	40408299984661.445273030317
b	6356752.314245179497563966

24.7.2 On the EGM96 Ellipsoid

The white noise RMS on azimuth and elevation measurements for the AFSCF is about 0.03 deg. Compare this to the values in the third column to see the disaster that would result from ignoring Earth's oblateness in ODTK IOD, particularly in the neighborhood of $\varphi = 45$ deg.

φ (deg)	φ' (deg)	$(\varphi - \varphi')$ (deg)	p (er)
0	0.0	0.0	1.0
5	4.9666960870092	0.033303912990772	0.99997474293723
10	9.9343942102791	0.065605789720866	0.99989972026697
15	14.904067139653	0.095932860347192	0.99977715704490
20	19.876629862375	0.12337013762532	0.99961069373012
25	24.852912560483	0.14708743951726	0.99940528551795
30	29.833635809829	0.16636419017093	0.99916706403359
35	34.819388702350	0.18061129765039	0.99890316414677
40	39.810610551928	0.18938944807156	0.99862151961304
45	44.807576784018	0.19242321598195	0.99833063226197
50	49.810389526291	0.18961047370932	0.99803932050022
55	54.818973309214	0.18102669078556	0.99775645392176
60	59.833076150493	0.16692384950734	0.99749068174564
65	64.852276137025	0.14772386297540	0.99725016354492
70	69.875993436424	0.12400656357643	0.99704231120188
75	74.903507474004	0.096492525996069	0.99687355113999
80	79.933978809966	0.066021190034306	0.99674911558149
85	84.966475056686	0.033524943313807	0.99667287082508
90	90.0	0.0	0.99664718933525

Chapter 25

Range

Range is *time delay* of a particular radio wave front from transmitter to receiver. Let t_T denote time of transmission, and t_R denote time of receipt. Then intrinsically, range $(t_R - t_T)$ has units of time. But typically and ideally, the time delay is multiplied by the speed of light c in a vacuum to define range $R_{SR} = c(t_R - t_T)$ with units of distance.

The radio wave front encounters high altitude clouds of electrons (ionosphere) between transmitter and receiver. Each electron interacts with the wave front with an energy exchange in which the electron is displaced from its equilibrium position and oscillates at its resonance frequency, and the radio wave front suffers a delay Δt_I . The size of Δt_I depends on the radio wave length, or frequency. At S-BAND (2 GHz) Δt_I is significant. At Ku-BAND (18 GHz) Δt_I is relatively insignificant. When range measurements are generated at two distinct frequencies the first order effect of Δt_I can be easily removed. This is the case with L-BAND GPS range measurements.

If the transmitter and/or receiver is located at a ground station, then the radio wave front encounters low altitude neutral tropospheric atoms and molecules (dominated by nitrogen and oxygen). The refracted wave front suffers tropospheric refraction and a delay Δt_R . Tropospheric refraction is independent of carrier frequency but varies significantly with elevation of antenna. The current version of ODTK uses the 'RC [106] model developed for the AFSCN.

When a transponder sits between transmitter and receiver there always exists a transponder time delay Δt_T . It is usually measured (with measurement error) prior to use of the transponder, but varies as a function of its thermal environment.

There are many other range delays ($\Delta t_O =$ sum of other delays) with varying degrees of significance. Unmodelled cable lengths generate time delays. Variations in the thermal and humidity environment of ground cables are responsible for variations in time delay. Antenna electromagnetic phase center varies as a function of geometry. Rotation between transmitter antenna and receiver antenna generates a time delay (Faraday rotation).

Each of the two times t_T and t_R is defined by a clock (oscillator). If t_T

and t_R are defined by the same clock, the range measurements are said to be coherent. If t_T and t_R are defined by distinct clocks, the range measurements are said to be non-coherent.

Two Clocks

Consider first the case for non-coherent range measurements; *i.e.*, the case for two distinct clocks. There are two significant and independent clock frequency effects that must be modeled: relativity and clock phenomenology.

According to Albert Einstein, if these two clocks have different velocities with respect to the same inertial frame, then they will tick at different frequencies (special relativity); clock frequency is a function of speed. And if they are at different distances from the central body center of mass, then they will tick at different frequencies (general relativity); *i.e.*, clock frequency is a function of gravitational field. Both special and general relativity must be deterministically modeled.

When separate clocks are used for transmitter and receiver, it is usually the case that one is inferior to the other. That is, the fractional frequency stability of one clock is superior. Then when the time t_R of one clock is compared to the time t_T of the other to define R_{SR} , the frequency and phase errors of the inferior clock are exposed. These random clock errors must be modeled by stochastic processes.

We associate the term *one-way* range measurement with two clocks.

One Clock

When the transmitter clock and receiver clock are the same clock, then the speed and location of transmitter clock and receiver clock are the same. The frequency effects due to special and general relativity vanish. Also, the train of radio waves received are a mirror image of those emitted, so random clock frequency and phase errors generated by the transmitter are subtracted out by the receiver. Clock phenomenology vanishes. Coherent range measurements are thus relatively easy to model. We associate the term *two-way* range measurement with one clock.

There can be one or more transponders between the colocated transmitter and receiver. Transponders are placed on spacecraft and/or at ground locations (*e.g.*, NASA TDRSS). Thus *two-way* range measurements may have two or more legs with one or more transponders, but only one clock.

Range measurement time-tag is physically associated with the clock used to define the time-tag. Possible candidates are t_T and t_R . Experience suggests that t_R is preferred. Note that transponder times of receipt and emission are not accessible by the transmitter/receiver clock.

25.1 Classical Two-Way Range

Classical two-way range refers to use of a transmitter/receiver, and its clock, at a ground station, and a transponder on a spacecraft.. Time-node order for classical two-way range: $t_i < t_j < t_p < t_q$.

25.1.1 Notation

c speed of light in vaccum

Uplink

t_i time of radio wave emission from ground station radar
 \mathbf{s}_i ECI position vector of ground station radar at time t_i
 t_j time of radio wave receipt at spacecraft transponder
 \mathbf{r}_j ECI position vector of spacecraft at time t_j
 \mathbf{R}_{ij} range vector, function of t_i and t_j
 R_{ij} one-leg range from radar to spacecraft
 f_{UT} radar transmitter frequency

Transponder Bias

β_T transponder time bias

Downlink

t_p time of radio wave emission from spacecraft transponder
 \mathbf{r}_p ECI position vector of spacecraft at time t_p
 t_q time of radio wave receipt at ground station radar
 \mathbf{s}_q ECI position vector of ground station radar at time t_q
 \mathbf{R}_{pq} range vector, function of t_p and t_q
 R_{pq} one-leg range from spacecraft to radar
 f_{DT} transponder transmitter frequency

25.1.2 Definitions

Uplink

$$\mathbf{R}_{ij} = \mathbf{r}_j - \mathbf{s}_i \quad (25.1)$$

Note that range vector \mathbf{R}_{ij} points to node with latest time $t_j > t_i$.

$$R_{ij} = \sqrt{\mathbf{R}_{ij} \cdot \mathbf{R}_{ij}} \quad (25.2)$$

$$\hat{\mathbf{R}}_{ij} = \mathbf{R}_{ij} / R_{ij} \quad (25.3)$$

Transponder

$$\beta_T = t_q - t_p \quad (25.4)$$

Downlink

$$\mathbf{R}_{pq} = \mathbf{s}_q - \mathbf{r}_p \quad (25.5)$$

Comparison of Eq. 25.5 with Eq. 25.1 identifies a convention in which the range vector always points to the node with latest time. \mathbf{R}_{pq} points to node at time $t_q > t_p$. This convention produces symmetry in the one-way equations for range-rate.

$$R_{pq} = \sqrt{\mathbf{R}_{pq} \cdot \mathbf{R}_{pq}} \quad (25.6)$$

$$\hat{\mathbf{R}}_{pq} = \mathbf{R}_{pq}/R_{pq} \quad (25.7)$$

Two-Way

$$R_{iq} = R_{ij} + c\beta_T + R_{pq} \quad (25.8)$$

25.1.3 Special Relativity

$$R_{ij} = c(t_j - t_i) \quad (25.9)$$

$$R_{pq} = c(t_q - t_p) \quad (25.10)$$

We now have two expressions for range R_{ij} and two expressions for range R_{pq} . Eq. 25.2 defines R_{ij} geometrically in a flat cartesian space, and Eq. 25.9 presents a derived result for R_{ij} from special relativity theory. Similarly for R_{pq} .

25.1.4 Calculation

Notice that the range vector \mathbf{R}_{ij} is not an instantaneous vector. Thus the range $R_{ij} = \sqrt{\mathbf{R}_{ij} \cdot \mathbf{R}_{ij}}$ is not invariant to rotation: One obtains different evaluations in inertial and Earth fixed components. Second, spherical radio waves referred to an inertial frame are not spherical when transformed to an Earth fixed rotating frame; *i.e.*, the straight-line propagation of radio plane waves in an inertial frame transform to curved-line propagation in an Earth fixed rotating frame. Bottom line: Make life easy by performing all related evaluations using inertial vector components, and remember to independently rotate the station vector from Earth-fixed to inertial components twice; *i.e.*, at both transmit and receive times. This paragraph is dedicated to Dr Jim Woodburn.

The simultaneous satisfaction of Eqs. 25.2 and 25.9 defines an uplink scalar nonlinear problem that must be solved by iteration. Similarly for downlink. One of the four time-nodes for two-way range $t_i < t_j < t_p < t_q$ is the given

two-way range time-tag t_R . Begin calculation at the time-tag t_R , and solve the problem for uplink and downlink with the transponder bias inserted between. Example: If $t_R = t_q$ (this is typical), solve the downlink problem first.

In units of time, the relativistic two-way range representation is given by:

$$R_{ij}/c + \beta_T + R_{pq}/c$$

In units of distance, the relativistic component of the range representation is given by:

$$R_{SR} = R_{ij} + c\beta_T + R_{pq} \quad (25.11)$$

at time-tag t_R .

25.1.5 Complete Representation

Let us denote the complete two-way range measurement representation by R . Then:

$$R = R_{SR} + R_R + R_I + R_{GM} + w_R \quad (25.12)$$

where R_R is tropospheric refraction, R_I is the ionospheric perturbation, R_{GM} is a sequentially correlated Gauss-Markov random variable designed to accommodate the sum of all other unmodeled range effects, and w_R is white Gaussian thermal noise. Parameter w_R is not observable, and is not estimated. The parameter R_{GM} is observable, and is estimated by the sequential filter.

25.2 Satellite to Satellite Two-Way Range

Satellite to satellite two-way range refers to use of a transmitter/receiver, and its clock, on one spacecraft (the tracker), and a transponder on another spacecraft (the target). There is perfect symmetry between classical two-way ranging and satellite to satellite two-way ranging. For the latter, the tracker spacecraft with the transmitter/receiver and its clock is a moving "ground station", and the target spacecraft with the transponder is the spacecraft with the transponder. With this correspondence, the classical two-way ranging is directly applicable to satellite to satellite two-way ranging.

25.3 Bi-Static Range (One-Way)

Bi-static ranging refers to use of a transmitter/clock combination at a transmitting facility, a relay transponder on tracked spacecraft (the target), and a receiver/clock combination at a receiving facility. The bi-static range measurement is analogous to classical two-way ranging, except the transmitter and receiver are not necessarily co-located, and the uplink path length may be significantly different than the downlink path. Each link path, or "leg", is treated

as a one-way ranging measurement, and the transponder delay is applied to the downlink portion of the measurement. In the modeling of bi-static ranging, ODTK assumes that the transmitter clock and the receiver clock are completely synchronized, and therefore does not account for stochastic clock errors in either the uplink or downlink range measurements.

Chapter 26

Doppler

The Doppler measurement, defined by the receiver, is characterized by a *phase count* N_{hm} on the Doppler frequency across a particular phase count time interval. Since the transmit frequency is known, the Doppler frequency can be extracted from the observed radio carrier frequency by differencing the observed frequency with the transmit frequency. Let t_h and t_m denote the initial and final times of the phase count time interval $[t_h, t_m]$, and define $\Delta t = (t_m - t_h)$. These times are defined by the receiver clock and the counting is performed by the receiver. Intrinsically the Doppler measurement consists of an integer (integer phase count), plus a fraction, of accumulated phase, where each integer count is defined by a 2π excursion on phase. If the receiver presents the phase count N_{hm} , the Doppler measurement has units of cycles. If the receiver presents $N_{hm}/\Delta t$ rather than N_{hm} , the Doppler measurement has units of cycles/time (mean frequency). If the receiver presents cN_{hm}/f_T , the Doppler measurement has units of distance (range-difference). If the receiver presents $cN_{hm}/(f_T\Delta t)$, the Doppler measurement has units of distance/time (mean range-rate).

The Doppler measurement is perturbed by ionospheric Total Electron Content (TEC) in the signal path. This effect must be modeled.

It is useful to think of the Doppler measurement in terms of its representation as a range-difference. If the range is perturbed by a particular physical effect, then the range difference over distinct times is perturbed accordingly. If a particular physical effect is a time-constant across light-time delay intervals within the phase count interval, then it will difference to zero. But this is usually *not* the case.

The Doppler measurement time-tag is physically associated with the clock used to define the phase count time interval. It is customary to place the measurement time-tag at the time t_m at end of the phase count time interval.

Note that many sequential range values may be calculated within the phase count time interval. When both range and Doppler measurements are derived and presented by the receiver, it is customary to present only one range measurement with one Doppler measurement where both have the same time tag.

26.1 Classical Two-Way Doppler

26.1.1 Notation

λ	wave length of radio carrier
ω_D	Doppler angular frequency
f_D	Doppler frequency
f_O	observed frequency

Uplink

\dot{R}_{ij}^j	range-rate, due to relative motion between radar and spacecraft
t_i^j	frequency shift
n_{ij}	number of wave lengths in uplink one-way signal path
φ_{ij}	total angular excursion by radio wave from radar to spacecraft

Downlink

\dot{R}_{pq}^q	range-rate, due to relative motion between radar and spacecraft
t_p^q	frequency shift
n_{pq}	number of wave lengths in downlink one-way signal path
φ_{pq}	total angular excursion by radio wave from spacecraft to radar

Two-Way

φ_{iq}	total angular excursion by radio wave from radar to spacecraft to radar
\dot{R}_{iq}^q	two-way range-rate, due to relative motion between radar and spacecraft
Δt_{hm}	radio carrier phase count time interval
N_{hm}	phase count on Doppler frequency

26.1.2 Definitions

$$f_T = c/\lambda \quad (26.1)$$

$$\omega_D = d\varphi/dt \quad (26.2)$$

$$f_D = \omega_D / (2\pi) \quad (26.3)$$

$$N_{hm} = \int_{t_h}^{t_m} f_D dt_q \quad (26.4)$$

$$\Delta R_{hm} = \int_{t_h}^{t_m} dR = R(t_m) - R(t_h) \quad (26.5)$$

$$f_O = f_T + f_D \quad (26.6)$$

$$M_{hm} = \int_{t_h}^{t_m} f_O dt_q \quad (26.7)$$

$$\Delta t = t_m - t_h \quad (26.8)$$

Uplink

$$\dot{R}_{ij}^j = dR_{ij}/dt_j \quad (26.9)$$

$$\dot{t}_i^j = dt_i/dt_j \quad (26.10)$$

$$n_{ij} = R_{ij}/\lambda \quad (26.11)$$

$$\varphi_{ij} = -2\pi n_{ij} \quad (26.12)$$

Downlink

$$\dot{R}_{pq}^q = dR_{pq}/dt_q \quad (26.13)$$

$$\dot{t}_p^q = dt_p/dt_q \quad (26.14)$$

$$n_{pq} = R_{pq}/\lambda \quad (26.15)$$

$$\varphi_{pq} = -2\pi n_{pq} \quad (26.16)$$

Two-Way

$$n_{iq} = R_{iq}/\lambda \quad (26.17)$$

$$\varphi_{iq} = -2\pi n_{iq} \quad (26.18)$$

26.1.3 Expressions for Uplink Range-Rate

Differentiate Eq. 25.1 with respect to time t_j :

$$\frac{d\mathbf{R}_{ij}}{dt_j} = \frac{d\mathbf{r}_j}{dt_j} - \frac{d\mathbf{s}_i}{dt_i} \frac{dt_i}{dt_j}$$

That is:

$$\dot{\mathbf{R}}_{ij}^j = \dot{\mathbf{r}}_j - \dot{\mathbf{s}}_i \dot{t}_i^j \quad (26.19)$$

Combine Eqs. 25.2 and 25.9 to write:

$$R_{ij} = \sqrt{\mathbf{R}_{ij} \cdot \mathbf{R}_{ij}} = c(t_j - t_i) \quad (26.20)$$

Differentiate Eq. 26.20 with respect to time t_j , and insert Eq. 26.19:

$$\dot{R}_{ij}^j = \hat{\mathbf{R}}_{ij} \cdot (\dot{\mathbf{r}}_j - \dot{\mathbf{s}}_i t_i^j) = c(1 - \dot{t}_i^j) \quad (26.21)$$

Use the equation with second and third expressions from Eq. 26.21 to solve for \dot{t}_i^j :

$$\dot{t}_i^j = \frac{1 - \hat{\mathbf{R}}_{ij} \cdot \dot{\mathbf{r}}_j / c}{1 - \hat{\mathbf{R}}_{ij} \cdot \dot{\mathbf{s}}_i / c} \quad (26.22)$$

Use the equation with first and third expressions from Eq. 26.21 to calculate \dot{R}_{ij}^j :

$$\dot{R}_{ij}^j = c(1 - \dot{t}_i^j) \quad (26.23)$$

Solve Eq. 26.23 for the frequency shift \dot{t}_i^j :

$$\dot{t}_i^j = 1 - \dot{R}_{ij}^j / c \quad (26.24)$$

Alternate Expressions

Expand Eq. 26.22 to write:

$$\dot{t}_i^j = 1 - \left(\frac{\hat{\mathbf{R}}_{ij} \cdot (\dot{\mathbf{r}}_j - \dot{\mathbf{s}}_i)}{c} \right) \left[1 + \frac{\hat{\mathbf{R}}_{ij} \cdot \dot{\mathbf{s}}_i}{c} + \frac{(\hat{\mathbf{R}}_{ij} \cdot \dot{\mathbf{s}}_i)^2}{c^2} + \dots \right] \quad (26.25)$$

and:

$$\dot{R}_{ij}^j = \left(\hat{\mathbf{R}}_{ij} \cdot (\dot{\mathbf{r}}_j - \dot{\mathbf{s}}_i) \right) \left[1 + \frac{\hat{\mathbf{R}}_{ij} \cdot \dot{\mathbf{s}}_i}{c} + \frac{(\hat{\mathbf{R}}_{ij} \cdot \dot{\mathbf{s}}_i)^2}{c^2} + \dots \right] \quad (26.26)$$

On comparison to Eq. 26.21, Eq. 26.26 provides a numerically stable method for evaluation of \dot{R}_{ij}^j .

26.1.4 Expressions for Downlink Range-Rate

Differentiate Eq. 25.5 with respect to time t_q :

$$\frac{d\mathbf{R}_{pq}}{dt_q} = \frac{d\mathbf{s}_q}{dt_q} - \frac{d\mathbf{r}_p}{dt_p} \frac{dt_p}{dt_q}$$

That is:

$$\dot{\mathbf{R}}_{pq}^q = \dot{\mathbf{s}}_q - \dot{\mathbf{r}}_p t_p^q \quad (26.27)$$

Combine Eqs. 25.6 and 25.10 to write:

$$R_{pq} = \sqrt{\mathbf{R}_{pq} \cdot \mathbf{R}_{pq}} = c(t_q - t_p) \quad (26.28)$$

Differentiate Eq. 26.28 with respect to time t_q , and insert Eq. 26.27:

$$\dot{R}_{pq}^q = \hat{\mathbf{R}}_{pq} \cdot (\dot{\mathbf{s}}_q - \dot{\mathbf{r}}_p t_p^q) = c(1 - \dot{t}_p^q) \quad (26.29)$$

Use the equation with second and third expressions from Eq. 26.29 to solve for \dot{t}_p^q :

$$\dot{t}_p^q = \frac{1 - \hat{\mathbf{R}}_{pq} \cdot \dot{\mathbf{s}}_q / c}{1 - \hat{\mathbf{R}}_{pq} \cdot \dot{\mathbf{r}}_p / c} \quad (26.30)$$

Use the equation with first and third expressions from Eq. 26.29 to calculate \dot{R}_{pq}^q :

$$\dot{R}_{pq}^q = c(1 - \dot{t}_p^q) \quad (26.31)$$

Solve Eq. 26.31 for the frequency shift \dot{t}_p^q :

$$\dot{t}_p^q = 1 - \dot{R}_{pq}^q / c \quad (26.32)$$

Alternate Expressions

Expand Eq. 26.30 to write:

$$\dot{t}_p^q = 1 - \left(\frac{\hat{\mathbf{R}}_{pq} \cdot (\dot{\mathbf{s}}_q - \dot{\mathbf{r}}_p)}{c} \right) \left[1 + \frac{\hat{\mathbf{R}}_{pq} \cdot \dot{\mathbf{r}}_p}{c} + \frac{(\hat{\mathbf{R}}_{pq} \cdot \dot{\mathbf{r}}_p)^2}{c^2} + \dots \right] \quad (26.33)$$

and:

$$\dot{R}_{pq}^q = (\hat{\mathbf{R}}_{pq} \cdot (\dot{\mathbf{s}}_q - \dot{\mathbf{r}}_p)) \left[1 + \frac{\hat{\mathbf{R}}_{pq} \cdot \dot{\mathbf{r}}_p}{c} + \frac{(\hat{\mathbf{R}}_{pq} \cdot \dot{\mathbf{r}}_p)^2}{c^2} + \dots \right] \quad (26.34)$$

26.1.5 Two-Way Range-Rate

Recall Eq. 25.8:

$$R_{iq} = R_{ij} + c\beta_T + R_{pq}$$

If the time-tag is at time t_q , then differentiate with respect to t_q :

$$\frac{dR_{iq}}{dt_q} = \frac{dR_{ij}}{dt_j} \frac{dt_j}{dt_p} \frac{dt_p}{dt_q} + c \frac{d\beta_T}{dt_q} + \frac{dR_{pq}}{dt_q} \quad (26.35)$$

By definition:

$$\beta_T = t_q - t_p$$

so:

$$d\beta_T = dt_q - dt_p$$

Since the transponder bias β_T is independent of Doppler frequency, we have:

$$d\beta_T = 0$$

and this implies:

$$dt_p = dt_j$$

Then Eq. 26.35 reduces to:

$$\frac{dR_{iq}}{dt_q} = \frac{dR_{ij}}{dt_q} + \frac{dR_{pq}}{dt_q} \quad (26.36)$$

where:

$$\frac{dR_{ij}}{dt_q} = \frac{dR_{ij}}{dt_j} \frac{dt_j}{dt_p} \frac{dt_p}{dt_q} \quad (26.37)$$

and:

$$\frac{dt_j}{dt_p} = 1 \quad (26.38)$$

26.1.6 Two-Way Doppler Frequency Equation

Recall that φ_{iq} is the total angular excursion by radio wave from radar to spacecraft to radar. Insert Eq. 26.17 into Eq. 26.18:

$$\varphi_{iq} = -2\pi R_{iq}/\lambda \quad (26.39)$$

and then insert Eq. 26.1:

$$\varphi_{iq} = -2\pi f_T R_{iq}/c \quad (26.40)$$

Differentiate Eq. 26.40 with respect to time t_q :

$$\frac{d\varphi_{iq}}{dt_q} = \frac{-2\pi f_T}{c} \frac{dR_{iq}}{dt_q} \quad (26.41)$$

and use Eqs. 26.2 and 26.3 to get the two-way Doppler frequency equation:

$$f_D = -\frac{f_T}{c} \frac{dR_{iq}}{dt_q} \quad (26.42)$$

26.1.7 Two-Way Phase Count Equation

Take definite time integrals over both sides of Eq. 26.42 between t_h and t_m :

$$\int_{t_h}^{t_m} f_D dt_q = -\frac{f_T}{c} \int_{t_h}^{t_m} dR_{iq} \quad (26.43)$$

and use the definition for N_{hm} to get the phase count equation:

$$N_{hm} = -\frac{f_T}{c} [R_{iq}(t_m) - R_{iq}(t_h)] \quad (26.44)$$

where N_{hm} is the two-way radio carrier phase count, f_T is transmitter frequency produced by the ground station clock, $R_{iq}(t_h)$ is the two-way range at beginning of the phase count interval and $R_{iq}(t_m)$ is the two-way range at the end of the phase count time interval $[t_h, t_m]$.

The left hand quantity N_{hm} is produced only by receiver hardware. The right-hand expression $-f_T \Delta R_{hm}/c$ is used for the Doppler measurement representation in orbit determination algorithms. Eq. 26.44 relates receiver hardware on the left to orbit determination software on the right.

The radio wave propagation time interval $[t_i, t_q]$ is typically very short compared to the arbitrary phase count time interval $[t_h, t_m]$. Check to see that both the left-hand and right-hand sides of Eq. 26.44 have units of cycles; *i.e.*, phase count.

Various forms of Eq. 26.44 are generated by manipulating f_T and c . Then the Doppler measurement will have different units. For all of these cases N_{hm} is produced by the receiver and ΔR_{hm} is used for the Doppler measurement representation in orbit determination algorithms.

Times t_q do not exist continuously with real t . Thus the Riemann integrals given in Eq. 26.43 should be replaced with discreet sums, Dirac-delta sifting integrals, or appropriate Lebesgue integrals. The Riemann notation is symbolic.

26.1.8 Observed Frequency

Uplink at Spacecraft with Time t_j

From definitions and Eq. 26.42 we have the frequency equation for f_{Oj} measured by an observer in motion (*e.g.*, a spacecraft) relative to the transmitter:

$$f_{Oj} = f_T \left(1 - \dot{R}_{ij}^j/c\right) \quad (26.45)$$

where the factor $\left(1 - \dot{R}_{ij}^j/c\right)$ is the uplink frequency shift.

Downlink at Radar with Time t_q

If transponder performs no frequency shift, then:

$$f_{Oq} = f_{Oj} \left(1 - \dot{R}_{pq}^q/c\right) = f_T \left(1 - \left(\dot{R}_{ij}^j + \dot{R}_{pq}^q\right)/c + \dot{R}_{ij}^j \dot{R}_{pq}^q/c^2\right) \quad (26.46)$$

26.1.9 Measurement Names

Receiver hardware engineers (*e.g.*, Motorola) quite naturally refer to N_{hm} as a phase count measurement. DMA (NIMA) engineers have referred to phase count measurements as Doppler measurements because the Doppler frequency f_D is integrated with time to produce phase count. Rearrange Eq. 26.44:

$$-\frac{c}{f_T} N_{hm} = \Delta R_{hm} \quad (26.47)$$

If the receiver produces values of $-N_{hm}c/f_T$, it is easy to see why some engineers (NSWC) have referred to these measurements as delta-range measurements ΔR_{hm} . Divide this equation by $\Delta t = (t_m - t_h)$:

$$-\frac{c}{f_T} \frac{N_{hm}}{\Delta t} = \frac{\Delta R_{hm}}{\Delta t} \quad (26.48)$$

and we have the same units as range-rate $\Delta R_{hm}/\Delta t$ (AFSCF). But to model a measurement derived from phase count as a range-rate is a big mistake, because $|\Delta t|$ is never arbitrarily small. Divide Eq. 26.44 through by Δt to get:

$$\frac{N_{hm}}{\Delta t} = -\frac{f_T}{c} \frac{\Delta R_{hm}}{\Delta t} \quad (26.49)$$

If the receiver (*e.g.*, NASA TDRSS) produces values of $N_{hm}/\Delta t$, then the measurements have frequency units, and some engineers refer to them as frequency measurements.

To summarize, we must address various transmitter/receiver hardware implementations with various names, but with common physics. For all of these cases N_{hm} is produced by the receiver and ΔR_{hm} is used for the Doppler measurement representation in orbit determination algorithms.

26.2 SCF ARTS Doppler

'Automated Remote Tracking Station' (ARTS) is used to identify any ground tracking station in the Air Force Satellite Control Facility (AFSCF).

Multiply Eq. 26.44 through by $-c/(f_T(t_m - t_h))$ to get:

$$-\frac{c}{f_T} \frac{N_{hm}}{(t_m - t_h)} = \frac{R_{iq}(t_m) - R_{iq}(t_h)}{(t_m - t_h)} \quad (26.50)$$

where:

$$(t_m - t_h) = 1 \text{ sec} \quad (26.51)$$

The left-hand side of Eq. 26.50 describes the Doppler measurement presented by the ARTS receiver, and the right-hand side describes the Doppler representation to be used for orbit determination software. $R_{iq}(t_m)$ and $R_{iq}(t_h)$ are both two-way ranges with uplink-downlink time nodes: $t_i < t_j < t_p < t_q$. The ARTS

Doppler measurement is inappropriately referred to as a range-rate measurement by ARTS documentation.

Note that a value for c is required to solve for ranges $R_{iq}(t_m)$ and $R_{iq}(t_h)$ by orbit determination software. This raises the question as to whether the same value for c is used on both sides of Eq. 26.50.

26.2.1 Complete Representation

Let ψ denote the complete two-way Doppler measurement representation. Then:

$$\psi = \psi_{SR} + \psi_I + \psi_R + \psi_{GM} + w_\psi \quad (26.52)$$

where ψ_{SR} is a function of spacecraft position and velocity, ψ_I is due to ionosphere, ψ_R is due to troposphere, ψ_{GM} is a sequentially correlated Gauss-Markov random variable designed to accommodate the sum of all other unmodeled Doppler effects, and w_ψ is Gaussian white noise. Parameter w_ψ is not observable, and is not estimated. The parameter ψ_{GM} is observable, and is estimated by the sequential filter.

26.2.2 Partial Derivatives

Position and Velocity

Recall Eq. 1.35, notation for spacecraft position and velocity:

$$Z = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ Z_3 \\ Z_4 \\ Z_5 \\ Z_6 \end{bmatrix}$$

Let us denote the right-hand side of Eq. 26.50 with ψ_{SR} :

$$\psi_{SR} = \frac{R_{iq}(t_m) - R_{iq}(t_h)}{(t_m - t_h)} \quad (26.53)$$

Differentiate ψ_{SR} with respect to $Z(t_m)$, where $t_m = t_q$, to construct the 1×6 matrix of partials:

$$\frac{\partial \psi}{\partial Z(t_m)} = \frac{\partial \psi_{SR}}{\partial Z(t_m)} = \frac{1}{(t_m - t_h)} \left[\frac{\partial R_{iq}(t_m)}{\partial Z(t_m)} - \frac{\partial R_{iq}(t_h)}{\partial Z(t_h)} \frac{\partial Z(t_h)}{\partial Z(t_m)} \right] \quad (26.54)$$

remembering that the difference $(t_m - t_h)$ is a time constant. The 6×6 orbit transition matrix:

$$\Phi_Z(t_h, t_m) = \frac{\partial Z(t_h)}{\partial Z(t_m)}$$

moves the denominator of the 1×6 matrix $\partial R_{iq}(t_h)/\partial Z(t_h)$ from time t_h to time t_m . Velocity partials emerge due to this transition matrix. Identify position and velocity partials:

$$\frac{\partial R_{iq}(t_\alpha)}{\partial Z(t_\alpha)} = \left[\frac{\partial R_{iq}(t_\alpha)}{\partial z(t_\alpha)}, \frac{\partial R_{iq}(t_\alpha)}{\partial \dot{z}(t_\alpha)} \right], \quad \alpha \text{ in } \{h, m\} \quad (26.55)$$

Then:

$$\frac{\partial R_{iq}(t_\alpha)}{\partial z(t_\alpha)} = (L_x, L_y, L_z) \quad (26.56)$$

$$\frac{\partial R_{iq}(t_\alpha)}{\partial \dot{z}(t_\alpha)} = (0, 0, 0) \quad (26.57)$$

using approximations:

$$\hat{\mathbf{R}}_{pq}(t_\alpha) = (L_x, L_y, L_z) [\mathbf{i}] \quad (26.58)$$

Doppler Bias

$$\frac{\partial \psi}{\partial \beta_\psi} = 1 \quad (26.59)$$

26.3 NASA STDN Doppler

26.3.1 Observed Range Rate

The NASA GSFC STDN document [103] defines the *Observed Range Rate*. The Doppler measurement is the cumulative cycle count of 1000 times the Doppler frequency plus a 240-MHz bias frequency. It is time tagged at the time of cycle counter reading. To process, convert Doppler data to decimal form. The observed average range rate is:

$$\dot{R}(T_0) = \frac{-c}{2f_T KM} \left[\frac{N(T_0) - N(T_{-1})}{T_0 - T_{-1}} - 2.4 \times 10^8 \right] \quad (26.60)$$

where:

c	speed of light
f_T	transmit frequency in Hertz
K	240/221
M	1000
N	cumulative Doppler-plus-bias counter reading
T_0, T_{-1}	time of present and previous Doppler count, respectively

26.3.2 Doppler Representation from the Observed Range Rate Equation

We shall attempt to derive an appropriate Doppler representation model equation from the observed range-rate model Eq. 26.60 (and Teles' equation (4-2) [107]) in terms of the notation given previously herein. Define the S-BAND input frequency f_{in} with:

$$f_{in}(t) = f_D(t) + 240\text{KHz} \quad (26.61)$$

where $f_D(t)$ denotes the Doppler frequency. Define the output Doppler count $N(t_m) - N(t_h)$ over $[t_h, t_m]$ with:

$$N(t_m) - N(t_h) = \int_{t_h}^{t_m} J f_{in}(t) dt \quad (26.62)$$

where:

$$J = KM \quad (26.63)$$

Then:

$$N(t_m) - N(t_h) = J \left\{ \int_{t_h}^{t_m} f_D(t) dt + (240\text{KHz})(t_m - t_h) \right\} \quad (26.64)$$

Define the average Doppler frequency $\bar{f}_D(t_m)$ with:

$$\bar{f}_D(t_m) = \frac{\int_{t_h}^{t_m} f_D(t) dt}{t_m - t_h} \quad (26.65)$$

and use Eq. 26.64 to get:

$$\bar{f}_D(t_m) = \frac{1}{KM} \left\{ \frac{N(t_m) - N(t_h)}{t_m - t_h} - 2.40 \times 10^8 \right\} \text{Hz} \quad (26.66)$$

Multiply through both equations by $-c/(2f_T)$ to get:

$$\left(\frac{-c}{2f_T} \right) \bar{f}_D(t_m) = \left(\frac{-c}{2f_T} \right) \left(\frac{\int_{t_h}^{t_m} f_D(t) dt}{t_m - t_h} \right) = \left(\frac{-c}{2f_T KM} \right) \left\{ \frac{N(t_m) - N(t_h)}{t_m - t_h} - 2.40 \times 10^8 \right\} \text{Hz} \quad (26.67)$$

Compare Eq. 26.67 with Eq. 26.60 to write:

$$\dot{R}(t_m) = \left(\frac{-c}{2f_T} \right) \left(\frac{\int_{t_h}^{t_m} f_D(t) dt}{t_m - t_h} \right) \quad (26.68)$$

Insert Eq. 26.43 to get:

$$\dot{R}(t_m) = \left(\frac{1}{2}\right) \left(\frac{R_{iq}(t_m) - R_{iq}(t_h)}{t_m - t_h}\right) \quad (26.69)$$

Eq. 26.69 is a one-way Doppler representation model due to the factor of 1/2. Since we model two-way range and Doppler internally as two-way measurements our implementation of Eq.26.69 is given by:

$$\left[\dot{R}(t_m)\right]_{2\text{-way}} = \frac{R_{iq}(t_m) - R_{iq}(t_h)}{t_m - t_h} \quad (26.70)$$

Chapter 27

Troposphere

Our use of the term *troposphere range* refers to *tropospheric propagation delay* in distance units, or *range error due to tropospheric refractivity*. Troposphere range is one additive component of the complete one-way range measurement between a ground station and a spacecraft.

27.1 Troposphere Range

Let R denote troposphere range. Let ε denote the geometric (non-refracted) elevation angle of an instantaneous range vector, referred to the local tangent plane, and let θ denote the refracted elevation (apparent elevation). It is conventional to indicate the dependence of troposphere range R on the geometric elevation ε : $R \equiv R(\varepsilon)$. Current computational methods have two very different functions in common: (i) A function to calculate the zenith troposphere range component $R(\pi/2)$, and (ii) A multiplicative mapping function $m(\varepsilon, \varepsilon_Z) = m(\varepsilon, \pi/2)$ that maps the zenith troposphere range component $R(\pi/2)$ to other elevations ε . Thus:

$$R(\varepsilon) = m(\varepsilon, \pi/2) R(\pi/2) \quad (27.1)$$

27.1.1 Zenith Component

Physics

Refractivity N is related to index of refraction n with:

$$N = 10^{-6} (n - 1) \quad (27.2)$$

For a spherically symmetric atmosphere, an integral expression for troposphere range $R(\pi/2)$ at zenith¹ has been presented (Mendes and Langley[83] Eq. (4)):

¹Langley[59] also presents an integral expression for approximate troposphere range as a function of non-zenith elevation.

$$R(\pi/2) = 10^{-6} \int_{r_S}^{r_A} N dr \quad (27.3)$$

where r_S is the geocentric radius of the Earth's surface and r_A is the geocentric radius of top of the neutral atmosphere.

Crane Model (not implemented in ODTK)

The classic model for radio refractivity of the neutral troposphere was defined by Crane ([15]1976):

$$N = k_1 \frac{p_d}{T} + k_2 \frac{p_w}{T} + k_3 \frac{p_w}{T^2} \quad (27.4)$$

where p_d (mb) and p_w (mb) are partial pressures of dry gases and water vapor, T is absolute temperature (K), and:

$$\begin{aligned} k_1 & 77.6 \pm 0.1K/mb \\ k_2 & 72.0 \pm 9.0K/mb \\ k_3 & (3.75 \pm 0.03) \times 10^5 K^2/mb \end{aligned} \quad (27.5)$$

Values for p_d (mb) and p_w (mb) are derived from measurements of temperature, pressure, and relative humidity. See below.

Thayer/Owens Improvement (not implemented in ODTK)

Crane's model was improved by Thayer ([108]1974²) to account for non-ideal gaseous behavior of the atmosphere:

$$N = k_1 \frac{p_d}{T} Z_d^{-1} + k_2 \frac{p_w}{T} Z_w^{-1} + k_3 \frac{p_w}{T^2} Z_w^{-1} \quad (27.6)$$

where Z_d^{-1} and Z_w^{-1} are inverse compressibility factors for dry air and water vapor. Expressions for them are given by Owens ([92]1967³):

$$Z_d^{-1} = 1.0 + 10^{-4} p_d \left(57.90 \times 10^{-4} \left(1.0 + \frac{0.52}{T} \right) - 9.4611 \frac{t}{T^2} \right) \quad (27.7)$$

$$Z_w^{-1} = 1.0 + 1650.0 \times 10^{-5} \frac{p_w}{T^3} (1.0 \times 10^4 - 1317.0t + 17.5t^2 + 0.144t^3) \quad (27.8)$$

where T and t are temperatures in units Kelvin and Celsius. Constants for Eq. 27.6:

$$\begin{aligned} k_1 & 77.6036 \pm 0.014K/mb \\ k_2 & 64.8000 \pm 0.080K/mb \\ k_3 & (3.7760 \pm 0.004) \times 10^5 K^2/mb \end{aligned} \quad (27.9)$$

²Thayer's extension seems to predate Crane. The Crane date was given by Ifadis without reference, and Ifadis implies that Thayer's extension is an improvement over Crane. Langley's bibliography[59] confirms the two dates.

³Indeed!?

Davis Modification

A more compact form for Eq. 27.6 was presented by Davis, et. al. ([17]1985):

$$N = k_1 R_d \rho + k_2' \frac{p_w}{T} Z_w^{-1} + k_3 \frac{p_w}{T^2} Z_w^{-1} \quad (27.10)$$

using equation of state:

$$p_i = Z_i \rho_i R_i T \quad (27.11)$$

where p_i is the partial pressure of an atmospheric constituent, ρ_i is the mass density, and R_i the specific gas constant for that constituent. So R_d is the specific gas constant for dry air, and $k_2' = (17 \pm 10) K/mb$.

Davis also presents the form of the hydrostatic delay based on Saastemoinen,

$$R_h = \left[(0.0022768 \pm 0.0000005) \frac{m}{mbar} \right] \frac{P_0}{f(\lambda, H)} \quad (27.12)$$

where

$$f(\lambda, H) = 1 - 0.00266 \cos(2\lambda) - 0.00028H \quad (27.13)$$

In the above, λ is the geodetic latitude and H is the height in kilometers above the geoid of the reference point. This is the a priori zenith delay model used in the Saastemoinen model in ODTK.

Refractivity Partition

Refractivity is presented as the sum of hydrostatic and wet components:

$$N = N_h + N_w \quad (27.14)$$

where:

$$N_h = k_1 R_d \rho \quad (27.15)$$

$$N_w = \left(k_2 \frac{p_w}{T} + k_3 \frac{p_w}{T^2} \right) Z_w^{-1} \quad (27.16)$$

Tropospheric Range Partition

Tropospheric range has the form:

$$R = R_h + R_w \quad (27.17)$$

where:

$$R_h = 10^{-6} \int_{r_S}^{r_A} N_h ds \quad (27.18)$$

$$R_w = 10^{-6} \int_{r_S}^{r_A} N_w ds \quad (27.19)$$

27.1.2 Mapping Function

All mapping functions tested by Mendes and Langley[83] demonstrated sub-centimeter accuracy for elevation angles above 15 degrees. For elevation angles below 10 degrees, only a select few were found to adequately meet the requirements imposed by space geodetic techniques. Functions by Herring, Ifadis, and Niel were recommended as best performing, especially for lower elevations. Least satisfactory performance was associated with Hopfield-based functions and the Marini-Murray function.

Marini ([69]1971) defined a continued fraction on $\sin(\varepsilon)$ to model the mapping function $m(\varepsilon)$:

$$m(\varepsilon) = \frac{1}{\sin(\varepsilon) + \frac{a}{\sin(\varepsilon) + \frac{b}{\sin(\varepsilon) + c}}} \quad (27.20)$$

and an illustrative numerical example is presented. Marini's mapping function was adopted and modified by several investigators. In common, each model has three parameter functions a , b , and c , but they are defined differently by each investigator.

IFADIS-GLOBAL (Ifadis) (not implemented in ODTK)

Marini's[69] model structure of the mapping function $m(\varepsilon)$ was adopted by Ifadis ([45]1986):

$$m(\varepsilon) = \frac{1}{\sin(\varepsilon) + \frac{a}{\sin(\varepsilon) + \frac{b}{\sin(\varepsilon) + c}}} \quad (27.21)$$

where:

$$a = k1 + k2(P_0 - 1000) + k3(T_0 - 15) + k4\sqrt{pw} \quad (27.22)$$

$$b = k1 + k2(P_0 - 1000) + k3(T_0 - 15) + k4\sqrt{pw} \quad (27.23)$$

Note: One of the repeated equations above is surely in error.

$$c = 0.078 \quad (27.24)$$

MTT (Herring) (not implemented in ODTK)

Marini's[69] model of the mapping function $m(\eta)$ was modified by Herring ([35]1992) to arrive at the form:

$$m(\varepsilon) = \frac{1 + \frac{a}{b}}{\sin(\varepsilon) + \frac{1+c}{a}} \frac{1}{\sin(\varepsilon) + \frac{b}{\sin(\varepsilon) + c}} \quad (27.25)$$

where[44]:

$$a = [1.2320 + 0.0130 \cos \varphi - 0.0209 H_\sigma + 0.00215 (T_s - 10)] 10^{-3} \quad (27.26)$$

$$b = [3.1612 - 0.1600 \cos \varphi - 0.0331 H_s + 0.00206 (T_s - 10)] 10^{-3} \quad (27.27)$$

$$c = [71.244 - 4.293 \cos \varphi - 0.149 H_s - 0.0021 (T_s - 10)] 10^{-3} \quad (27.28)$$

NMF Niell

Niell (27.301993) extended the Herring formula with a height correction term:

$$m(\varepsilon) = \frac{1 + \frac{a}{b}}{\sin(\varepsilon) + \frac{1+c}{a}} \frac{1}{\sin(\varepsilon) + \frac{b}{\sin(\varepsilon) + c}} + 10^{-3} H_s M_h \quad (27.29)$$

where:

$$M_h = \frac{1}{\sin \varepsilon} - \frac{1 + \frac{a_{ht}}{b_{ht}}}{\sin(\varepsilon) + \frac{1 + c_{ht}}{a_{ht}}} \frac{1}{\sin(\varepsilon) + \frac{b_{ht}}{\sin(\varepsilon) + c_{ht}}} \quad (27.30)$$

and a_{ht} , b_{ht} , and c_{ht} are constants. See Niell ([90]2002) for his recent model of mapping functions.

The parameters (a, b, c) of the hydrostatic mapping function all have the form

$$\alpha(\lambda, t) = \alpha_{avg}(\lambda) - \alpha_{amp}(\lambda) \cos\left(2\pi \frac{t - T_0}{365.25}\right) \quad (27.31)$$

where $\alpha_{avg}(\lambda)$ and $\alpha_{amp}(\lambda)$ are computed based on interpolation of tabulated values as a function of geodetic latitude (λ), t is the time in days past the

beginning of the year and T_0 is a phase of 28 days. The parameters (a, b, c) of the wet mapping function are simply constants which are interpolated from tabulated values as a function of geodetic latitude. These are the mapping functions used with the Saastemoinen model in ODTK. The combination of the Saastemoinen a priori hydrostatic zenith model with the Niell NMF mapping function is also the recommended by McCarthy and Petit [72].

Marini Murray

The Marini Murray model is commonly used in satellite laser ranging applications. It is not separated into hydrostatic and wet component, but takes the form

$$R = \frac{f(\lambda)}{f(\varphi, H)} \frac{A + B}{\sin E + \frac{B/(A+B)}{\sin E + 0.01}} \quad (27.32)$$

The site location function is given by

$$f(\varphi, H) = 1 - 0.00266 \cos(2\varphi) - 0.00031H \quad (27.33)$$

where φ is the geodetic latitude and H is the height in kilometers above the geoid of the reference point. The laser frequency parameter is given by

$$f(\lambda) = 0.9650 + \frac{0.0164}{\lambda^2} + \frac{0.000228}{\lambda^4} \quad (27.34)$$

where λ is the laser wavelength in micrometers. A and B are computed as

$$A = 0.002357P_0 + 0.000141e_0 \quad (27.35)$$

$$B = (1.084 \times 10^{-8}) P_0 T_0 K + (4.734 \times 10^{-8}) \left[\frac{P_0^2}{T_0} \right] \left[\frac{2}{3 - 1/K} \right] \quad (27.36)$$

where,

$$e_0 = \left[\frac{(\text{relative humidity } (\%))}{100} \right] e_S f_W \quad (27.37)$$

$$e_S = 0.01 \exp$$

$$(1.2378847 \times 10^{-5} T_0^2 - 1.9121316 \times 10^{-2} T_0 + 33.93711047 - 6.3431645 \times 10^3 T_0^{-1}) \quad (27.38)$$

$$f_W = 1.00062 + 3.14 \times 10^{-6} P_0 + 5.6 \times 10^{-7} (T_0 - 273.15)^2 \quad (27.39)$$

$$K = 1.163 - 0.00968 \cos(2\varphi) - 0.00104T_0 + 0.00001435P_0 \quad (27.40)$$

and T_0 is the temperature in degrees Kelvin, P_0 is the pressure in millibars and (relative humidity (%)) is the relative humidity expressed as a percentage. The use of this model with laser ranging data is the recommended by McCarthy and Petit [72].

Estimation Estimation of corrections to the troposphere corrections is done estimating corrections to the zenith delay of the wet component. The hydrostatic portion of the troposphere correction is assumed to be perfectly accurate. The contribution of the wet component is initially zero. The partial of the troposphere range correction with respect to the troposphere estimate is therefore given by the wet mapping function.

27.2 Partial Pressure Measurements

A partial pressure is the pressure that would be exerted by one component of a mixture of gases if it were present alone. The total atmospheric pressure is the sum of all partial pressures. Partial pressure values are derived from in-situ measurements of atmospheric pressure, temperature, and relative humidity. The partial pressure p_d for dry gases is approximated with the measured total ambient pressure. The partial pressure p_w for water vapor is temperature dependent; *e.g.*, $p_w = 32mb$ for $T = 25 \text{ deg } C$ (tropics), and $p_w = 1.2mb$ for $T = -20 \text{ deg } C$ (polar vortex). Its value is derived from a measurement of relative humidity (RH):

$$p_w = (p_w^0) (RH)$$

where p_w^0 is the partial pressure of water vapor at saturation, derived from a temperature dependent table. And RH , for constant water vapor, is temperature dependent: If the amount of water vapor is held constant, then increasing temperature decreases RH , and decreasing temperature increases RH . If temperature is held constant, then adding water vapor increases RH , and removing water vapor decreases RH .

The dew-point is the temperature to which the air would have to be cooled, without changing pressure or moisture content, to reach saturation.

The psychrometer and chilled-mirror hygrometer (dew-point meter) are standard sensors for measurements of humidity. A carbon resistance hygrometer and a human-hair hygrometer have also been developed to measure humidity.

The psychrometer consists of two thermometers, a dry-bulb thermometer and a wet-bulb thermometer, and tables of associated data. The wet-bulb thermometer measures the temperature to which objects are cooled by evaporation of water. The temperature difference between dry-bulb and wet-bulb thermometers is indexed to the dry-bulb temperature in a standard table to derive a value of RH .

The chilled-mirror hygrometer is a temperature controlled mirror in which the mirror is cooled until condensation occurs. The condensation temperature

is the dew point measurement. The dew point is directly related to the amount of moisture in the air, so is an excellent indicator of absolute humidity.

The carbon resistance hygrometer derives from the observation that as humidity increases, water molecules fill up pores in a carbon resistor; *i.e.*, resistance in carbon is inversely proportional to relative humidity.

The human-hair hygrometer derives from the fact that human hair lengthens as relative humidity increases.

27.3 Atmospheric Thermodynamics

27.3.1 State Variables

Pressure

Atmospheric pressure p is the weight of the atmosphere per unit area. Average sea level pressure:

$$1atm = 1013.25mb = 14.7psi = 29.92''Hg$$

Temperature

Let T denote temperature, and let K , C , and F denote units in Kelvin, Centigrade, and Fahrenheit. Temperature is directly proportional to kinetic energy of a molecule. Molecular motion is zero at $T = 0K$.

$$T(K) = T(C) + 273.2K$$

$$T(C) = [5/9][T(F) - 32F]$$

Density

Air density ρ is mass of air per unit volume. Dry air at sea level and $273.2K$ has a density of $1.29kg/m^3$.

Specific Humidity

Specific humidity q is the ratio of the mass of water vapor to that of air. When the air is saturated, denote q with q_s . q_s is temperature dependent. Warmer air holds more water.

$$q \geq q_s$$

27.3.2 Ideal Gas Law

Consider a closed container of gas molecules colliding with container walls in accordance with Newton's laws. Let us denote:

n	number of moles in container
R	universal gas constant ($8.3145 J/molK$)
N	number of molecules in container
$k = R/N_A$	Boltzmann constant ($1.38066 \times 10^{-23} J/K$)
N_A	Avogadro's number (6.0221×10^{23})
p	absolute pressure in container
V	volume of container
T	temperature in container
$\rho = m/V$	molecular density in container
$R^* = nR$	

Then the ideal gas law can be written:

$$p = \frac{nRT}{V} = \frac{NkT}{V} = \frac{R^*}{m} \rho T$$

27.3.3 Relative Humidity and Partial Pressures

Let us denote the relative humidity with RH . Let p_w denote the actual partial pressure of water vapor, and let p_w^0 denote the saturation partial pressure of water vapor, both at the same temperature. Then[114]:

$$RH = \frac{p_w}{p_w^0} = \frac{q}{q_s}$$

Water vapor density (concentration) is a function of relative humidity and temperature. It increases smoothly with increasing relative humidity, and with increasing temperature.

Evaporation:

- Is a change in liquid water to vapor
- Is a cooling process
- Will occur as long as the air is not saturated
- Will stop when air is saturated

Saturation:

- Is defined when the total number of water molecules condensed from the air is balanced with the total number of water molecules evaporated into the air
- Can be realized by cooling air without adding or removing moisture, or by adding water vapor without changing temperature

- Is marked by a wet-bulb thermometer temperature
- Is measured by a psychrometer

Saturation vapor pressure:

- Is defined as the maximum vapor pressure that is thermodynamically stable for a given temperature
- Is the pressure that water vapor molecules would exert if the air were saturated at a given temperature
- Describes how much water vapor is necessary to make the air saturated at a given temperature
- Increases with temperature

Dew Point

- Is the temperature T_d at which the saturation vapor pressure equals the actual vapor pressure

A dew-point meter has a cooled surface. The air in contact with it is cooled at constant pressure until it saturates and begins to deposit dew. This is the dew-point temperature T_d . Since the amount of vapor in the air has not changed, and the pressure has remained constant, the partial pressure of water must be constant during the cooling process. Therefore (Wencsek[114]) :

$$p_w(T) = p_w^0(T_d)$$

$$RH = \frac{p_w^0(T_d)}{p_w^0(T)}$$

The measurement of RH by a dew-point meter is superior to that using a wet-bulb thermometer because the former invokes only condensation, whereas the latter invokes both condensation and evaporation – they act against each other. Also, the wet-bulb variation (depression) is only about half of the dew-point meter variation.

27.4 Troposphere Range Error

Let us denote any a priori modeled estimate of R with \bar{R} , and define the error in \bar{R} with:

$$\Delta R = R - \bar{R} \tag{27.41}$$

Define the relative error in \bar{R} with:

$$x = \left(\frac{\Delta R}{R} \right) \quad (27.42)$$

where ΔR is a random variable. The random variable x has a Gauss-Markov transition $\Phi_{k+1,k}$:

$$x_{k+1} = \Phi_{k+1,k} x_k + \sqrt{1 - \Phi_{k+1,k}^2} w_{x,k+1} \quad (27.43)$$

with elevation ε (nonlinear with time):

$$\Phi_{k+1,k} = \exp[\alpha(\varepsilon_{k+1} - \varepsilon_k)] \quad (27.44)$$

$\alpha < 0$, an input constant

where $w_{x,k+1}$ is an unbiased Gaussian white sequence and:

$$x_0 = w_{x,0} \quad (27.45)$$

$$\sigma_{x,k}^2 = E\{w_{x,k}^2\}, \text{ an input constant for all } k \quad (27.46)$$

27.4.1 Propagation Variance for Relative Troposphere Range Error

Square Eq. 27.43 and take expectation:

$$E\{x_{k+1}^2\} = \Phi_{k+1,k}^2 E\{x_k^2\} + (1 - \Phi_{k+1,k}^2) E\{w_{x,k+1}^2\} \quad (27.47)$$

For $k = 0$, using Eq. 27.45:

$$E\{x_0^2\} = E\{w_{x,0}^2\} = \sigma_{x,0}^2 \quad (27.48)$$

and using Eq. 27.47:

$$\begin{aligned} E\{x_1^2\} &= \Phi_{1,0}^2 E\{x_0^2\} + (1 - \Phi_{1,0}^2) E\{w_{x,1}^2\} \\ &= \Phi_{1,0}^2 E\{w_{x,0}^2\} + (1 - \Phi_{1,0}^2) E\{w_{x,1}^2\} \\ &= \sigma_{x,0}^2 \end{aligned}$$

Use Eq. 27.47 recursively with induction on the non-negative integers to show:

$$E\{x_k^2\} = \sigma_{x,0}^2, \quad k \in \{2, 3, \dots\} \quad (27.49)$$

That is, x_k is a stationary sequence on elevation ε . But from Eq. 27.42:

$$E\{x_k^2\} = \frac{E\{(\Delta R_k)^2\}}{R_k^2} = \frac{\sigma_{\Delta R_k}^2}{R_k^2} \quad (27.50)$$

So:

$$\sigma_{\Delta R_k}^2 = R_k^2 \sigma_{x,0}^2 \quad (27.51)$$

Since $\sigma_{x,0}^2$ is constant, then $\sigma_{\Delta R_k}^2$ increases if and only if R_k^2 increases, and $\sigma_{\Delta R_k}^2$ decreases if and only if R_k^2 decreases. From Eq. 27.25, R_k^2 decreases if and only if ε increases, and R_k^2 increases if and only if ε decreases.

Therefore: When the elevation ε increases, then error variance on troposphere range $\sigma_{\Delta R_k}^2$ decreases. And when ε decreases, then $\sigma_{\Delta R_k}^2$ increases. We thus have the desired inverse proportionality. These variations in $\sigma_{\Delta R_k}^2$ with ε are quantified by user input control constants for α and $\sigma_{x,0}$.

27.4.2 Sequential Estimation

The relative error x_j in \bar{R}_j is unknown. Given range measurements, run the conditional expectation operator through Eq. 27.43, and use Sherman's Theorem, to get:

$$\hat{x}_{k+1|k} = \Phi_{k+1,k} \hat{x}_{k|k} \quad (27.52)$$

because $w_{x,k+1}$ is zero mean. Define the error in $\hat{x}_{k|k}$ with:

$$\delta \hat{x}_{k|k} = x_k - \hat{x}_{k|k} \quad (27.53)$$

where x_k is unknown truth. Insert Eq. 27.43 into Eq. 27.53 to get:

$$\delta \hat{x}_{k+1|k} = \Phi_{k+1,k} \delta \hat{x}_{k|k} + \sqrt{1 - \Phi_{k+1,k}^2} w_{x,k+1} \quad (27.54)$$

Insert $\delta \hat{x}_{k+1|k}$ into $E \left\{ (\delta \hat{x}_{k+1|k})^2 \right\}$:

$$E \left\{ (\delta \hat{x}_{k+1|k})^2 \right\} = \Phi_{k+1,k}^2 E \left\{ (\delta \hat{x}_{k|k})^2 \right\} + (1 - \Phi_{k+1,k}^2) \sigma_{x,0}^2 \quad (27.55)$$

The two terms on the right-hand side define filter covariance propagation and deweighting, respectively. Propagation here refers to propagation in elevation ($\varepsilon_{k+1}, \varepsilon_k$), not time.

27.5 SCF 'RC Model

27.5.1 Tropospheric Range Model ('RC)

The current version of ODTK uses the 'RC [106] model for R , developed for the AFSCN.

$$R = \frac{5.6 \times 10^{-6} \bar{N}}{(\sin \varepsilon + 0.06483)^{1.4}} \quad (27.56)$$

where R has units km , ε denotes elevation angle, and \bar{N} denotes mean surface refractivity (either a constant monthly average, or a value interpolated from a series of monthly averages).

Tropospheric Range Error Model

Let us denote the tropospheric refraction one-way range error with ΔR , and define tropospheric refraction two-way range error variance about zero mean with:

$$\sigma_{\Delta R}^2 = 2E \left\{ (\Delta R)^2 \right\} \quad (27.57)$$

where serial correlation in ΔR_R is assumed negligible. The two-way root-variance is modeled and calculated with:

$$\sigma_{\Delta R} = \sqrt{2}C |R|, \quad \text{with } 0 < C < 1 \quad (27.58)$$

where the ratio C is a user defined stored constant, and $|R|$ is derived dynamically from the 'RC algorithm. Thus $\sigma_{\Delta R}$ is modeled as some fraction of $|R|$.

27.5.2 Doppler

The right-hand side of Eq. 26.50 describes the Doppler representation to be used for orbit determination software. Notice that the Doppler measurement is modeled as a range difference divided by a time difference. This provides a prototype model, particularly for modeling the tropospheric contribution to the complete Doppler representation:

$$\psi_R = \frac{R(t_m) - R(t_h)}{(t_m - t_h)} \quad (27.59)$$

where R is tropospheric range, and $[t_h, t_m]$ is the Doppler count interval.

Doppler Error Model

Let us denote the tropospheric refraction one-way Doppler error with $\Delta\psi$, and define tropospheric refraction two-way Doppler error variance about zero mean with:

$$\sigma_{\Delta\psi}^2 = 2E \left\{ (\Delta\psi)^2 \right\} \quad (27.60)$$

where serial correlation in $\Delta\psi_R$ is assumed negligible. The two-way root-variance is modeled with:

$$\sigma_{\Delta\psi} = \sqrt{2} \left(\frac{\sigma_{\Delta\psi}}{|\psi|} \right) |\psi|, \quad \text{with } 0 < \left(\frac{\sigma_{\Delta\psi}}{|\psi|} \right) < 1 \quad (27.61)$$

where the ratio $(\sigma_{\Delta\psi}/|\psi|)$ is a user defined stored constant, and $|\psi|$ is derived dynamically, as above. Thus $\sigma_{\Delta\psi}$ is modeled as some fraction of $|\psi|$.

Chapter 28

Ionosphere

The ionosphere is that part of the Earth's atmosphere above sixty kilometers which, when in equilibrium, consists of an electrically neutral plasma of positive ions and free negative electrons. Electrons are stripped from atoms and molecules by the ionizing action of the sun's ultraviolet radiation. In this plasma the ions are much more massive than the electrons. Consequently the passage of a radio wave front through the plasma has a much more significant effect on the electrons than on the ions. Electrons are moved from their equilibrium positions and then oscillate at resonant frequency. In this energy exchange the radio wave front suffers a frequency dependent phase delay, and any range or Doppler measurement derived from the radio wave front suffers a frequency dependent measurement perturbation. The purpose of this chapter is to present expressions for range and Doppler ionospheric perturbations.

The current version of ODTK uses the IRI (International Reference Ionosphere) 2001 model for calculation of Total Electron Content (TEC).

28.1 Range

Let R_I denote the additive ionospheric component to range. Then:

$$R_I = a \left(\frac{N_T}{f_T^2} \right) \quad (28.1)$$

where f_T is transmitter frequency, N_T is the two-way Total Electron Content (TEC) measured in units of $TECU$ where:

$$1 \text{ } TECU = \frac{10^{16} \text{ electrons}}{m^2} \quad (28.2)$$

and:

$$a = 4.030816 \times 10^5 \frac{(m) (MHz^2)}{TECU} \quad (28.3)$$

This evaluation provides R_I in units of meters (m). Two-way TEC is approximated by doubling the one-way range TEC experienced on the downward leg.

28.2 Doppler

Let N_{hm}^I denote the perturbation in Doppler phase count due to ionospheric effects, and let $N_{T,h}$ and $N_{T,m}$ denote two-way TEC at beginning and end of the phase count interval; *i.e.*, at phase count times t_h and t_m . Then:

$$N_{hm}^I = \left(\frac{a}{c}\right) \left(\frac{N_{T,m} - N_{T,h}}{f_T}\right) \quad (28.4)$$

where f_T is radio transmitter frequency, a is defined above, and c is speed of light in vacuum.

28.2.1 SCF ARTS Doppler (Range-Rate)

Let ψ_I denote the SCF ARTS range-rate component due to ionosphere with dimension distance per time. Multiply Eq. 28.4 through by the factor $-c/(f_T(t_m - t_h))$ to get:

$$\psi_I = -\frac{cN_{hm}^I}{f_T(t_m - t_h)} = -\left(\frac{a}{f_T^2}\right) \left(\frac{N_{T,m} - N_{T,h}}{(t_m - t_h)}\right) \quad (28.5)$$

This evaluation provides ψ_I in units of meters per second (m/s).

Chapter 29

TDOA

Time Difference Of Arrival observations provide a measure of the difference in distance traveled along two distinct signal paths which originate from the same emitter and end at different receivers. ODTK processing of these measurements assumes coherency between the receivers so that clock differences between the receivers need not be modeled or estimated despite the one way nature of the observations. We will refer to the two paths as signal path 1 and signal path 2 and assume that the observation time tag is at the time of receipt of signal on path 1. The difference measurement is formed in sense of time of arrival on path 2 (TOA2) minus the time of arrival on path 1 (TOA1).

29.1 Ground-based TDOA (Ground to Spacecraft to Ground)

In the ground based TDOA measurement, the signal is generated by a static ground based emitter, is transponded by two relay satellites and received by static ground based receivers. ODTK modeling currently assumes the same location of the ground based receiver for both signal paths. Since the observation time tag is at signal receipt on path 1, modeling of the TDOA measurement requires the computation of the relative time of signal receipt on path 2. This computation is performed by working backwards from the time of receipt on path 1 to determine the time of emission and then working forwards along the path 2 to determine the time of receipt on path 2.

29.1.1 Ground-based TDOA light time algorithm

Let t_{R1} be the time of signal reception on path 1. The time of transmission from the relay satellite on path 1, t_{ST1} , is determined by modeling the light time delay due to special relativity and adding the delays due to tropospheric and ionospheric effects,

$$t_{ST1} = t_{R1} - \frac{1}{c} [R_{SR} + \Delta R_{tropo} + \Delta R_{iono}] \quad (29.1)$$

where R_{SR} is the one way range based on special relativity,

$$R_{SR} = |\mathbf{R}_{S1}(t_{ST1}) - \mathbf{R}_R(t_{R1})|, \quad (29.2)$$

ΔR_{tropo} is the tropospheric delay in terms of range, ΔR_{iono} is the ionospheric delay in terms of range, \mathbf{R}_{S1} is the inertial position of relay satellite on path 1 and \mathbf{R}_R is the inertial position of the ground based receiver. The time of receipt of the signal at the relay satellite on path 1 is simply given as the time of transmission minus the transponder delay at the relay on path 1, β_1 ,

$$t_{SR1} = t_{ST1} - \beta_1 \quad (29.3)$$

The time of signal transmission from the emitter is then computed by modeling the light time delay due to special relativity and adding the delays due to tropospheric and ionospheric effects,

$$t_E = t_{SR1} - \frac{1}{c} [R_{SR} + \Delta R_{tropo} + \Delta R_{iono}] \quad (29.4)$$

where R_{SR} is the one way range based on special relativity,

$$R_{SR} = |\mathbf{R}_{S1}(t_{SR1}) - \mathbf{R}_E(t_E)|, \quad (29.5)$$

and \mathbf{R}_E is the inertial position of the ground based emitter. This completes the backwards leg of the TDOA computation along path 1. We now move forward along path 2 to compute the time of signal receipt at relay on path 2,

$$t_{SR2} = t_E + \frac{1}{c} [R_{SR} + \Delta R_{tropo} + \Delta R_{iono}], \quad (29.6)$$

where R_{SR} is the one way range based on special relativity,

$$R_{SR} = |\mathbf{R}_{S2}(t_{SR2}) - \mathbf{R}_E(t_E)| \quad (29.7)$$

and \mathbf{R}_{S2} is the inertial position of receiving satellite on path 2. The time of transmission from the relay on path 2 is simply given as the time of receipt plus the transponder delay at the relay on path 2, β_2 ,

$$t_{ST2} = t_{SR2} + \beta_2. \quad (29.8)$$

The time of signal arrival at the receiver along path 2 is then computed by modeling the light time delay due to special relativity and adding the delays due to tropospheric and ionospheric effects,

$$t_{R2} = t_{ST2} + \frac{1}{c} [R_{SR} + \Delta R_{tropo} + \Delta R_{iono}], \quad (29.9)$$

where R_{SR} is the one way range based on special relativity,

$$R_{SR} = |\mathbf{R}_{S2}(t_{ST2}) - \mathbf{R}_E(t_{R2})| \quad (29.10)$$

29.1.2 Complete Representation

The complete representation of the ground based TDOA measurement is then given by

$$TDOA = (t_{R2} - t_{R1}) + TDOA_{GM} + \omega_{TDOA} \quad (29.11)$$

where $TDOA_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled TDOA effects and ω_{TDOA} is white Gaussian thermal noise.

29.2 Ground TDOA (Spacecraft to Ground)

In the ground TDOA measurement, the signal is generated by a space based emitter and received by static ground based receivers. The observation time tag is at signal receipt at receive station 2, therefore modeling of the TDOA measurement requires the computation of the relative time of signal receipt at receive station 1. This computation is performed by working backwards from the time of receipt at station 2 to determine the time of emission and then working forwards determine the time of receipt at station 1.

29.2.1 Ground TDOA light time algorithm

Let t_{R2} be the time of signal reception at receive station 2. The time of transmission from the satellite based emitter, t_E , is determined by modeling the light time delay due to special relativity and adding the delays due to tropospheric and ionospheric effects,

$$t_E = t_{R2} - \frac{1}{c} [R_{SR2} + \Delta R_{tropo} + \Delta R_{iono}] \quad (29.12)$$

where R_{SR2} is the one way range based on special relativity,

$$R_{SR2} = |\mathbf{R}_E(t_E) - \mathbf{R}_{R2}(t_{R2})|, \quad (29.13)$$

ΔR_{tropo} is the tropospheric delay in terms of range, ΔR_{iono} is the ionospheric delay in terms of range, \mathbf{R}_E is the inertial position of space based emitter and \mathbf{R}_{R2} is the inertial position of ground based receiver 2.

We now move forward to compute the time of signal receipt at ground receive station 1,

$$t_{R1} = t_E + \frac{1}{c} [R_{SR1} + \Delta R_{tropo} + \Delta R_{iono}], \quad (29.14)$$

where R_{SR1} is the one way range based on special relativity,

$$R_{SR1} = |\mathbf{R}_{R1}(t_{R1}) - \mathbf{R}_E(t_E)|, \quad (29.15)$$

and \mathbf{R}_{R1} is the inertial position of ground based receiver 2.

29.2.2 Complete Representation

The complete representation of the ground TDOA measurement is then given by

$$TDOA = (t_{R2} - t_{R1}) + TDOA_{GM} + \omega_{TDOA} \quad (29.16)$$

where $TDOA_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled TDOA effects and ω_{TDOA} is white Gaussian thermal noise.

29.3 Ground-based single differenced TDOA (Ground to Spacecraft to Ground)

For a ground-based single-differenced TDOA measurement, the signals generated by two static ground-based emitters are transponded by two relay satellites and received at a ground-based receiver. ODTK modeling assumes the terminal receiver location is the same for both signal paths contributing to the TDOA measurement. Differencing TDOA measurements effectively removes the down-link part of the TDOA observation since those paths are common to all signals. A differenced TDOA measurement will also be independent of relay transponder delays if the signals pass through the same transponder paths. The disadvantage is that much of the geometrical information will subtract out if the emitters are near each other, and the white noise on differenced measurements is increased. The observation time tag is at emitter 1 signal receipt on path 1 (Emitter 1 -> Relay 1 -> Receiver). The difference is formed in the sense of emitter 2 TDOA minus emitter 1 TDOA.

$$SDTDOA = (TDOA_{E2} - TDOA_{E1}) + SDTDOA_{GM} + \omega_{SDTDOA} \quad (29.17)$$

29.4 Space-based TDOA (Ground to Spacecraft)

In the space based TDOA measurement, the signal is generated by a static ground based emitter and received by two space based receivers. Since the observation time tag is at signal receipt on path 1, modeling of the TDOA measurement requires the computation of the relative time of signal receipt on path 2. This computation is performed by working backwards from the time of receipt on path 1 to determine the time of emission and then working forwards along the path 2 to determine the time of receipt on path 2. Note that the assumption of a time tag at time of receipt of signal at the satellite means that all satellite based delays must be removed from the observation prior to processing in ODTK.

29.4.1 Space-based TDOA light time algorithm

Let t_{SR1} be the time of signal reception on path 1. The time of transmission from the emitter is then computed by modeling the light time delay due to special relativity and adding the delays due to tropospheric and ionospheric effects,

$$t_E = t_{SR1} - \frac{1}{c} [R_{SR} + \Delta R_{tropo} + \Delta R_{iono}] \quad (29.18)$$

where R_{SR} is the one way range based on special relativity,

$$R_{SR} = |\mathbf{R}_{S1}(t_{SR1}) - \mathbf{R}_E(t_E)|, \quad (29.19)$$

ΔR_{tropo} is the tropospheric delay in terms of range, ΔR_{iono} is the ionospheric delay in terms of range, \mathbf{R}_{S1} is the inertial position of receiving satellite on path 1 and \mathbf{R}_E is the inertial position of the ground based emitter. This completes the backwards leg of the TDOA computation along path 1. We now move forward along path 2 to compute the time of signal receipt at relay on path 2,

$$t_{SR2} = t_E + \frac{1}{c} [R_{SR} + \Delta R_{tropo} + \Delta R_{iono}], \quad (29.20)$$

where R_{SR} is the one way range based on special relativity,

$$R_{SR} = |\mathbf{R}_{S2}(t_{SR2}) - \mathbf{R}_E(t_E)| \quad (29.21)$$

and \mathbf{R}_{S2} is the inertial position of receiving satellite on path 2.

29.4.2 Complete Representation

The complete representation of the space based TDOA measurement is then given by

$$TDOA = (t_{SR2} - t_{SR1}) + TDOA_{GM} + \omega_{TDOA} \quad (29.22)$$

where $TDOA_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled TDOA effects and ω_{TDOA} is white Gaussian thermal noise.

Chapter 30

FDOA

Frequency Difference Of Arrival observations provide a measure of the difference in Doppler shift along two distinct signal paths which originate from the same emitter and end at different receivers. ODTK processing of these measurements assumes coherency between the receivers so that clock differences between the receivers need not be modeled or estimated despite the one way nature of the observations. We will refer to the two paths as signal path 1 and signal path 2 and assume that the observation time tag is at the time of receipt of signal on path 1. The difference measurement is formed in sense of frequency of arrival on path 2 (FOA2) minus the frequency of arrival on path 1 (FOA1).

30.1 Ground-based FDOA (Ground to Spacecraft to Ground)

In the ground based FDOA measurement, the signal is generated by a static ground based emitter, is transponded by two relay satellites and received by static ground based receivers. ODTK modeling currently assumes the same location of the ground based receiver for both signal paths. The ground-based TDOA light time algorithm is used to generate the positions and velocities of all satellites and ground locations required to support the computation of frequency of arrival along each signal path.

30.1.1 Doppler shift

The received frequency at location 2 of a transmission from location 1 at frequency f_t is computed as

$$f_R = A(\mathbf{R}_1, \mathbf{V}_1, \mathbf{R}_2, \mathbf{V}_2) f_t \quad (30.1)$$

where,

$$A(\mathbf{R}_1, \mathbf{V}_1, \mathbf{R}_2, \mathbf{V}_2) = \sqrt{\frac{1 - \frac{\dot{\rho}_{1,2}}{c}}{1 + \frac{\dot{\rho}_{1,2}}{c}}} \quad (30.2)$$

$$\rho_{1,2} = |\mathbf{R}_2 - \mathbf{R}_1| \quad (30.3)$$

$$\dot{\rho}_{1,2} = \frac{(\mathbf{R}_2 - \mathbf{R}_1) \cdot (\mathbf{V}_2 - \mathbf{V}_1)}{\rho_{1,2}} \quad (30.4)$$

In ODTK, A is computed based on a Taylor series expansion in $\frac{\dot{\rho}_{1,2}}{c}$,

$$A(\mathbf{R}_1, \mathbf{V}_1, \mathbf{R}_2, \mathbf{V}_2) \cong 1 - \frac{\dot{\rho}_{1,2}}{c} + \frac{1}{2} \left(\frac{\dot{\rho}_{1,2}}{c} \right)^2 \quad (30.5)$$

30.1.2 Frequency of arrival

The frequency of arrival at the relay satellite for a signal of frequency f_E transmitted by the emitter is computed as

$$f_{SR} = A(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_S(t_{SR}), \mathbf{V}_S(t_{SR})) f_E \quad (30.6)$$

When the signal travels through the transponder, the frequency is subjected to additive frequency shift, Δf , to yield the frequency transmitted from the satellite,

$$F_{ST} = f_{SR} + \Delta f \quad (30.7)$$

The frequency of the signal observed at the receiving ground station is then computed as

$$f_R = A(\mathbf{R}_S(t_{ST}), \mathbf{V}_S(t_{ST}), \mathbf{R}_R(t_R), \mathbf{V}_R(t_R)) f_{ST} \quad (30.8)$$

30.1.3 Complete Representation

The complete representation of the ground based FDOA measurement is then given by

$$FDOA = (f_{R2} - f_{R1}) + FDOA_{GM} + \omega_{FDOA} \quad (30.9)$$

where $FDOA_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled FDOA effects and ω_{FDOA} is white Gaussian thermal noise.

30.2 Ground FDOA (Spacecraft to Ground)

In the ground FDOA measurement, the signal is generated by a space based emitter and received by static ground based receivers. The observation time tag is at signal receipt at receive station 2. The ground TDOA light time algorithm is used to generate the positions and velocities of the emitter satellite and ground locations required to support the computation of frequency of arrival along each signal path.

30.2.1 Frequency of arrival

The frequency of arrival at ground receive station 2 for a signal of frequency f_E transmitted by the emitter is computed as

$$f_{R2} = A(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_{R2}(t_{R2}), \mathbf{V}_{R2}(t_{R2})) f_E \quad (30.10)$$

The frequency of arrival at ground receive station 1 for a signal of frequency f_E transmitted by the emitter is computed as

$$f_{R1} = A(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_{R1}(t_{R1}), \mathbf{V}_{R1}(t_{R1})) f_E \quad (30.11)$$

30.2.2 Complete Representation

The complete representation of the ground FDOA measurement is then given by

$$FDOA = (f_{R2} - f_{R1}) + FDOA_{GM} + \omega_{FDOA} \quad (30.12)$$

where $FDOA_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled FDOA effects and ω_{FDOA} is white Gaussian thermal noise.

30.3 Ground-based single differenced FDOA (Ground to Spacecraft to Ground)

For a ground-based single-differenced FDOA measurement, the signals generated by two static ground-based emitters are transponded by two relay satellites and received at a ground-based receiver. ODTK modeling assumes the terminal receiver location is the same for both signal paths contributing to the FDOA measurement. Differencing FDOA measurements effectively removes the down-link part of the FDOA observation since those paths are common to all signals. A differenced FDOA measurement will also be independent of relay transponder delays if the signals pass through the same transponder paths. The disadvantage is that much of the geometrical information will subtract out if the emitters are near each other, and the white noise on differenced measurements is increased. The observation time tag is at emitter 1 signal receipt on path 1 (Emitter 1 ->

Relay 1 -> Receiver). The difference is formed in the sense of emitter 2 FDOA minus emitter 1 FDOA.

$$SDFDOA = (FDOA_{E2} - FDOA_{E1}) + SDFDOA_{GM} + \omega_{SDFDOA} \quad (30.13)$$

30.4 Space-based FDOA (Ground to Spacecraft)

In the space based FDOA measurement, the signal is generated by a static ground based emitter and received by two satellites. The space-based TDOA light time algorithm is used to generate the positions and velocities of all satellites and ground locations required to support the computation of frequency of arrival along each signal path. The Doppler shift algorithm is given in the section on ground-based FDOA.

30.4.1 Frequency of arrival

The frequency of arrival at the relay satellite for a signal of frequency f_E transmitted by the emitter is computed as

$$f_{SR} = A(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_S(t_{SR}), \mathbf{V}_S(t_{SR})) f_E \quad (30.14)$$

30.4.2 Complete Representation

The complete representation of the space based FDOA measurement is then given by

$$FDOA = (f_{SR2} - f_{SR1}) + FDOA_{GM} + \omega_{FDOA} \quad (30.15)$$

where $FDOA_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled FDOA effects and ω_{FDOA} is white Gaussian thermal noise.

Chapter 31

TDOA Dot

Time Difference Of Arrival Dot observations provide a measure of the rate of change of the difference in distance traveled along two distinct signal paths which originate from the same emitter and end at different receivers. ODTK processing of these measurements assumes coherency between the receivers so that clock differences between the receivers need not be modeled or estimated despite the one way nature of the observations. We will refer to the two paths as signal path 1 and signal path 2 and assume that the observation time tag is at the time of receipt of signal on path 1. The difference measurement is formed in sense of rate of arrival on path 2 (TOADot2) minus the rate of arrival on path 1 (TOADot1).

31.1 Ground-based TDOA Dot (Ground to Spacecraft to Ground)

In the ground based TDOA Dot measurement, the signal is generated by a static ground based emitter, is transponded by two relay satellites and received by static ground based receivers. ODTK modeling currently assumes the same location of the ground based receiver for both signal paths. The ground-based TDOA light time algorithm is used to generate the positions and velocities of all satellites and ground locations required to support the computation of frequency of arrival along each signal path.

31.1.1 Range rate

The rate of change of the distance between location 2 and location 1 at frequency f_t is computed as

$$\rho = |\mathbf{R}_2 - \mathbf{R}_1| \quad (31.1)$$

$$\dot{\rho}(\mathbf{R}_1, \mathbf{V}_1, \mathbf{R}_2, \mathbf{V}_2) = \frac{(\mathbf{R}_2 - \mathbf{R}_1) \cdot (\mathbf{V}_2 - \mathbf{V}_1)}{\rho} \quad (31.2)$$

31.1.2 Rate of change of time of arrival

The rate of change in the uplink distance between the emitter and a relay satellite is given by

$$\dot{t}_{SR} = \frac{1}{c} \left[\dot{R}_{SR} + \Delta \dot{R}_{tropo} + \Delta \dot{R}_{iono} \right] \quad (31.3)$$

which is approximated in ODTK as

$$\dot{t}_{SR} \cong \frac{1}{c} \left[\dot{R}_{SR} \right] = \dot{\rho}(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_S(t_{SR}), \mathbf{V}_S(t_{SR})) \quad (31.4)$$

where the rate of change of contributions due to the troposphere and ionosphere are ignored. The rate of change in the downlink distance between a relay satellite and the ground receiver is given by

$$\dot{t}_R = \dot{t}_{SR} + \dot{\beta} + \frac{1}{c} \left[\dot{R}_{SR} + \Delta \dot{R}_{tropo} + \Delta \dot{R}_{iono} \right] \quad (31.5)$$

which is approximated in ODTK as

$$\dot{t}_R \cong \dot{t}_{SR} + \frac{1}{c} \left[\dot{R}_{SR} \right] = \dot{t}_{SR} + \dot{\rho}(\mathbf{R}_S(t_{ST}), \mathbf{V}_S(t_{ST}), \mathbf{R}_S(t_R), \mathbf{V}_S(t_R)) \quad (31.6)$$

where the rate of change of contributions due to the transponder delay, the troposphere and ionosphere are ignored.

31.1.3 Complete Representation

The complete representation of the ground based TDOA Dot measurement is then given by

$$TDOADot = (\dot{t}_{R2} - \dot{t}_{R1}) + TDOADot_{GM} + \omega_{TDOADot} \quad (31.7)$$

where $TDOADot_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled TDOA Dot effects and $\omega_{TDOADot}$ is white Gaussian thermal noise.

31.2 Space-based TDOA Dot (Ground to Spacecraft)

In the space based TDOA Dot measurement, the signal is generated by a static ground based emitter and received by space based receivers. The space-based

TDOA light time algorithm is used to generate the positions and velocities of all satellites and ground locations required to support the computation of the rate of change of time of arrival along each signal path.

31.2.1 Rate of change of time of arrival

The rate of change in the uplink distance between the emitter and a relay satellite is given by

$$\dot{t}_{SR} = \frac{1}{c} \left[\dot{R}_{SR} + \Delta \dot{R}_{tropo} + \Delta \dot{R}_{iono} \right] \quad (31.8)$$

which is approximated in ODTK as

$$\dot{t}_{SR} \cong \frac{1}{c} \left[\dot{R}_{SR} \right] = \dot{\rho}(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_S(t_{SR}), \mathbf{V}_S(t_{SR})) \quad (31.9)$$

where the rate of change of contributions due to the troposphere and ionosphere are ignored.

31.2.2 Complete Representation

The complete representation of the space based TDOA Dot measurement is then given by

$$TDOADot = (\dot{t}_{SR2} - \dot{t}_{SR1}) + TDOADot_{GM} + \omega_{TDOADot} \quad (31.10)$$

where $TDOADot_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled TDOA Dot effects and $\omega_{TDOADot}$ is white Gaussian thermal noise.

Chapter 32

FDOA Dot

32.1 Ground FDOA Dot

In the ground FDOA Dot measurement, the signal is generated by a space based emitter and received by a pair of static ground based receivers. The observation time tag is at signal receipt at receive station 2. The ground TDOA light time algorithm is used to generate the positions, velocities and accelerations of the emitter satellite and ground locations required to support the computation of frequency of arrival rate along each signal path.

32.1.1 Frequency of arrival rate

The frequency of arrival rate at ground receive station 2 for a signal of frequency f_E transmitted by the emitter is computed as

$$\dot{f}_{R2} = \frac{d}{dt} A(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_{R2}(t_{R2}), \mathbf{V}_{R2}(t_{R2})) f_E \quad (32.1)$$

The rate of change of the frequency of arrival at ground receive station 1 for a signal of frequency f_E transmitted by the emitter is computed as

$$\dot{f}_{R1} = \frac{d}{dt} A(\mathbf{R}_E(t_E), \mathbf{V}_E(t_E), \mathbf{R}_{R1}(t_{R1}), \mathbf{V}_{R1}(t_{R1})) f_E \quad (32.2)$$

where,

$$A(\mathbf{R}_1, \mathbf{V}_1, \mathbf{R}_2, \mathbf{V}_2) \approx 1 - \frac{\ddot{\rho}_{1,2}}{c} + \frac{1}{2} \left(\frac{\ddot{\rho}_{1,2}}{c} \right)^2 \quad (32.3)$$

and

$$\ddot{\rho}_{1,2} = \frac{1}{\rho_{1,2}} ((\mathbf{R}_2 - \mathbf{R}_1) \cdot (\mathbf{A}_2 - \mathbf{A}_1) + (\mathbf{V}_2 - \mathbf{V}_1) \cdot (\mathbf{V}_2 - \mathbf{V}_1) - \dot{\rho}_{1,2}^2) \quad (32.4)$$

32.1.2 Complete Representation

The complete representation of the ground FDOA Dot measurement is then given by

$$FDOADot = \left(\dot{f}_{R2} - \dot{f}_{R1} \right) + FDOADot_{GM} + \omega_{FDOADot} \quad (32.5)$$

where $FDOADot_{GM}$ is a sequentially correlated Gauss-Markov random variable which is designed to accommodate the sum of all unmodeled FDOA rate effects and $\omega_{FDOADot}$ is white Gaussian thermal noise.

Chapter 33

DSN Range and Total Count Phase

Processing of observations from the NASA/JPL Deep Space Network is supported via observation models for two-way sequential range, two-way and three-way Doppler and two-way and three-way total count phase measurements. In the context of DSN measurement modeling, a two-way measurement involves a signal which is received by the same ground station from which it was transmitted. A three-way measurement involves a signal which is received by the different ground station than the one from which it was transmitted. Transmitted signals may be ramped or constant in frequency. The mathematical model for computed values for these measurement types follows Moyer [87] except as noted below. All measurement modeling is consistent with DSN operations using Block V receivers following the completion of the Network Simplification Program.

33.1 Sequential Range

The DSN sequential range measurement is an ambiguous range. The length of the unambiguous part of the observable is supplied in the tracking data file in conjunction with the observation value. During the simulation of measurements, the ambiguity is set to be 10^{99} which results in generation of unambiguous ranges.

33.2 Total Count Phase

The DSN total count phase measurement is a measure of total received cycles of the carrier from a specified epoch. The epoch for the start of the count is supplied in the tracking data file in conjunction with the observed value. Sequential total count phase measurements are differenced during the reading of

the tracking data file so that the resulting measurement processed in ODTK is the received cycles over the current count interval. This differencing operation makes the processing of total count phase observations equivalent to processing Doppler observations. While the differencing operation may reduce the strength of the total count phase measurement, it simplifies the processing of the measurement in the sequential filter. During the simulation of measurements, a constant transmit frequency associated with the transmitting ground station is used.

33.3 Doppler

The DSN Doppler measurement is a measure of total received cycles of the carrier over the specified Doppler count interval. The count interval is supplied in the tracking data file in conjunction with the observed value. The Doppler measurement differs slightly from the total count phase measurement in that the time tag on the Doppler measurement is placed in the middle of the count interval while the time tag on the total count phase measurement is placed at the end of the count interval. During the simulation of measurements, a constant transmit frequency associated with the transmitting ground station is used.

33.4 Antenna Corrections

Antenna correction effects are modeled based on Moyer.

33.5 Media Corrections

The corrections for effects of the troposphere and the ionosphere in Moyer are replaced with the troposphere and ionosphere models available in ODTK.

33.6 Solar Corona Model

The effects of the solar corona are not currently modeled.

Chapter 34

Clock Modeling

34.1 Introduction

The currently implemented ODTK three-state clock model was derived from that presented by Zucca-Tavella[105] (ZT, 2005) but can be shown to be functionally equivalent to the two-state model by David Allan[1] (DA, 1990) when the third state in the ZT model is ignored, and to the three state GEONS model. The three states in the ODTK model are referred to as clock phase deviation, frequency deviation and frequency drift (aging). The ODTK clock model is stochastically characterized by clock model diffusion constants σ_1 , σ_2 , and σ_3 as introduced by ZT. The exact relationship between clock model diffusion coefficients and the Allan variance was the objective of ZT (ZT page 290).

In the presentation of each model (Zucca-Tavella, Allan and GEONS) different notation is used to refer to common clock model parameters which can lead to some confusion. Table 34.1 presents a clock model notational map that relates the notation used in the ODTK user interface to the notation used in the description of ZT, DA and GEONS. Note that while the ODTK model follows ZT, the ODTK inputs are a combination of terms from Allan's presentation with frequency drift based on ZT. The **FM** acronym used three times in Table 34.1 header is read: *frequency modulated*.

	FM white noise	FM random walk	FM frequency drift
Zucca-Tavella	σ_1	σ_2	σ_3
Allan	$\sqrt{a_0}$	$\sqrt{3a_{-2}}$	NONE
GEONS	$\sqrt{\dot{Q}_{bR}}$	$\sqrt{\dot{Q}_{dR}}$	$\sqrt{\dot{Q}_{dR}}$
ODTK	$\sqrt{a_0}$	$\sqrt{3a_{-2}}$	σ_3

Table 34.1: Diffusion Coefficient Parameter Map

Row 1 of Table 34.1 presents ZT notation for clock parameter diffusion coefficients. Row 2 presents equivalent DA notation for clock parameter diffusion coefficients, but DA did not address FM frequency drift. Row 3 presents GEONS notation for clock parameter diffusion coefficients. Row 4 of Table 34.1 presents adopted ODTK notation for clock parameter diffusion coefficients. ODTK stochastic clock input parameters are a_0 , a_{-2} , and σ_3 .

The ODTK filter-smoother estimates the 3×1 matrix clock state $X(t)$ for each clock C_i according to Equations 34.42 through 34.49, where $Q_{k+1,k}$ is used for clock C_i filter process noise covariance, and the estimation of $X_3(t)$ (aging) is user optional. Allan parameter values for a_0 , a_{-2} , and ZT diffusion coefficient σ_3 are input to the filter by the user. Allan parameter values for a_0 and a_{-2} can be determined by the user by inspection of user Allan variance diagram for each clock C_i . The GEONS[32] deterministic relativistic clock phase drift Δd_{Rel}^n for each satellite clock is modeled in ODTK with user option selection.

34.2 Time

34.2.1 Variable t

Let t denote the instantaneous value of the mean phase of a superior ensemble of atomic clocks (examples: USNO ensemble, GPS ensemble), where t is referred to a particular origin. Let C_i denote a clock that is not a member of the atomic clock ensemble, and let $x(t)$ denote the instantaneous value of the phase (time) deviation of clock C_i from t . Let $y(t)$ denote the instantaneous value of the fractional frequency deviation of clock C_i referred to the mean frequency of the atomic clock ensemble.

34.2.2 Constant τ

For $k \in \{0, 1, 2, \dots\}$, let t_k and t_{k+1} denote estimates of t by clock C_i . For all k , define a clock C_i time length constant $\tau > 0$ by

$$\tau = t_{k+1} - t_k \quad (34.1)$$

34.3 ODTK Simulator Clock Time Update

The clock three-state simulated deviation $X(t_k)$ is propagated from t_k to t_{k+1}

$$X(t_{k+1}) = \Phi(t_{k+1}, t_k) X(t_k) + G(t_{k+1}) \quad (34.2)$$

with the 3×3 transition matrix

$$\Phi(t_{k+1}, t_k) = \begin{bmatrix} 1 & \tau & \tau^2/2 \\ 0 & 1 & \tau \\ 0 & 0 & 1 \end{bmatrix} \quad (34.3)$$

and the random functional $G(t_{k+1})$ defined by Equation 34.47 in Section 34.6.

34.4 ODTK Filter Clock Time Update

34.4.1 State

The clock three-state estimate $\hat{X}(t_k)$ is propagated from t_k to t_{k+1}

$$\hat{X}(t_{k+1}) = \Phi(t_{k+1}, t_k) \hat{X}(t_k) \quad (34.4)$$

with transition matrix defined by Equation 34.3.

34.4.2 Covariance

The clock three-state estimate error covariance $P(t_k)$ is propagated to time t_{k+1}

$$P(t_{k+1}) = \Phi(t_{k+1}, t_k) P(t_k) \Phi^T(t_{k+1}, t_k) + Q_{k+1,k} \quad (34.5)$$

with process noise covariance $Q_{k+1,k}$

$$Q_{k+1,k} = \begin{bmatrix} \sigma_1^2 \tau + \sigma_2^2 \frac{\tau^3}{3} + \sigma_3^2 \frac{\tau^5}{20} & \sigma_2^2 \frac{\tau^2}{2} + \sigma_3^2 \frac{\tau^4}{8} & \sigma_3^2 \frac{\tau^3}{6} \\ \sigma_2^2 \frac{\tau^2}{2} + \sigma_3^2 \frac{\tau^4}{8} & \sigma_2^2 \tau + \sigma_3^2 \frac{\tau^3}{3} & \sigma_3^2 \frac{\tau^2}{2} \\ \sigma_3^2 \frac{\tau^3}{6} & \sigma_3^2 \frac{\tau^2}{2} & \sigma_3^2 \tau \end{bmatrix} \quad (34.6)$$

34.5 Allan Variance

Initial values for clock parameters identified in Table 34.1 are useful for operators of spacecraft equipped with space qualified clocks. The clock manufacturer of each space qualified clock usually provides an associated Allan variance diagram. The spacecraft operator can estimate Allan parameter values of the two Allan parameters a_0 and a_{-2} from that Allan variance diagram. This provides initial values for two of the three clock parameters. Conversion to ZT and GEONS forms can then be achieved using Table 34.1.

Allan[1], Equation (6) page 649, defines the average fractional frequency \bar{y}_k for the k^{th} measurement interval

$$\bar{y}_k = \frac{x_{k+1} - x_k}{\tau} \quad (34.7)$$

where:

$$x_k = x(t_k) \quad (34.8)$$

Define

$$\Delta \bar{y} = \bar{y}_{k+1} - \bar{y}_k \quad (34.9)$$

The sample Allan variance is defined by Allan page 649 Equation (7)

$$\sigma_y^2(\tau) = \frac{1}{2} \langle (\Delta \bar{y})^2 \rangle \quad (34.10)$$

where $\langle \cdot \rangle$ indicates an infinite time average. Then

$$\sigma_y^2(\tau) = \frac{1}{2} \left\langle (\bar{y}_{k+1} - \bar{y}_k)^2 \right\rangle = \lim_{n \rightarrow \infty} \left[\frac{1}{2n} \sum_{k=1}^n (\bar{y}_{k+1} - \bar{y}_k)^2 \right] \quad (34.11)$$

Allan[1] (page 651 Table II) presents representations of $\sigma_y^2(\tau)$ for frequency modulated (FM) white noise, FM flicker noise, and FM random walk

$$\sigma_y^2(\tau) = \begin{cases} a_0 \tau^{-1} & \text{FM white noise} \\ a_{-1} & \text{FM flicker noise} \\ a_{-2} \tau & \text{FM random walk} \end{cases} \quad (34.12)$$

where a_0 , a_{-1} , and a_{-2} are C_i clock dependent constants. FM flicker noise is difficult to model or simulate, and is usually ignored. Allan also presents representations of $\sigma_y^2(\tau)$ for phase modulated (PM) white noise and PM flicker noise. But $\sigma_y^2(\tau)$ is dominated by the frequency modulated sequences that integrate into phase.

34.5.1 Covariance on Clock Phase

Values for clock dependent parameter constants a_0 and a_{-2} define selected models for composition of the Allan variance $\sigma_y^2(\tau)$ according to Eqs. 34.12. We assume that the clock can be sufficiently characterized¹ by frequency modulated (FM) white noise and frequency modulated random walk. Then clock phase modulation (PM) will be driven by their generalized integrals.

Let $x(t)$ denote clock time deviation (or clock phase deviation) and let $y(t)$ denote clock frequency deviation. Formally, Allan[1] presents the (generalized²) integral:

$$x(t) = \int_0^t y(\eta) d\eta \quad (34.13)$$

Frequency White Noise

The physical Allan variance for FM white noise is given by:

$$\sigma_y^2(\tau) = a_0 \tau^{-1}, \text{ FM white noise} \quad (34.14)$$

Compare Eqs. 34.11 and 34.14 to write:

¹This assumption is supported by Allan: "Note that the quadratic D term occurs because $x(t)$ is the integral of $y(t)$, the fractional frequency, and is often the predominant cause of time deviation."

²Allan does not distinguish between integrals and generalized integrals in his paper referenced above. But the standard calculus does not work for the "integration" of white noise and random walk sequences. These sequences are summable and differenceable, but they are not integrable and differentiable. It is nonetheless helpful to communicate the relation between x and y by analogy to the standard calculus in the absence of measure theory and Lebesgue integration tools.

$$a_0\tau^{-1} = \lim_{n \rightarrow \infty} \frac{1}{2n} \sum_{j=1}^n (\bar{y}_{j+1} - \bar{y}_j)^2 \quad (34.15)$$

From this equation derive:

$$E \{x(t_k) x(t_j)\} = (a_0) (\min [t_j, t_k]) \quad (34.16)$$

A candidate value: $a_0 = 10^{-22} s$.

Frequency Random Walk

The physical Allan variance for FM random walk is given by:

$$\sigma_y^2(\tau) = a_{-2}\tau, \text{ FM random walk} \quad (34.17)$$

Compare Eqs. 34.11 and 34.17 to write:

$$a_{-2}\tau = \lim_{n \rightarrow \infty} \frac{1}{2n} \sum_{j=1}^n (\bar{y}_{j+1} - \bar{y}_j)^2 \quad (34.18)$$

Adopt the reset to zero condition:

$$y_j = 0, t_j = 0 \quad (34.19)$$

Then derive:

$$E \left\{ (y(t))^2 \right\} = 3a_{-2}t \quad (34.20)$$

A candidate value: $a_{-2} = (10^{-25}/300) s^{-1}$.

Superposition

Ignoring FM flicker noise and phase modulated (PM) sequences given by Allan, then:

$$\sigma_y^2(\tau) = a_0\tau^{-1} + a_{-2}\tau \quad (34.21)$$

34.5.2 Simulations

FM White Noise

Let $t_0 = 0$, τ_0 a positive small time increment, x_0 a positive small number, and n an unbounded positive integer. Define:

$$t = n\tau_0 > t_0 \quad (34.22)$$

$$x(t) = \sum_{i=1}^n x_i \quad (34.23)$$

and adopt the probability statement:

$$P \{x_i = x_0\} = P \{x_i = -x_0\} = 0.5 \quad (34.24)$$

Then [93]:

$$E \left\{ (x(t))^2 \right\} = \left(\frac{x_0^2}{\tau_0} \right) t \quad (34.25)$$

when the sizes of τ_0 and x_0 are sufficiently small. From Eq. 34.16:

$$E \left\{ (x(t))^2 \right\} = a_0 t \quad (34.26)$$

Comparison of the latter two equations yields:

$$x_0 = \sqrt{a_0 \tau_0} \quad (34.27)$$

Given $t_0 = 0$, and values for a_0 and τ_0 , derive x_0 from Eq. 34.27, and with the help of a uniform random number generator, simulate the random walk sequence in clock phase, defined by Eq. 34.23, due to FM white noise. Derive the corresponding white noise sequence in clock frequency with the help of Eq. 34.7.

FM Random Walk

Let $t_0 = 0$, τ_0 a positive small time increment, y_0 a positive small number, and n an unbounded positive integer. Define:

$$t = n\tau_0 > t_0 \quad (34.28)$$

$$y(t) = \sum_{i=1}^n y_i \quad (34.29)$$

and adopt the probability statement:

$$P \{y_i = y_0\} = P \{y_i = -y_0\} = 0.5 \quad (34.30)$$

Then [93]:

$$E \left\{ (y(t))^2 \right\} = \left(\frac{y_0^2}{\tau_0} \right) t \quad (34.31)$$

when the sizes of τ_0 and y_0 are sufficiently small. From Eq. 34.20:

$$E \left\{ (y(t))^2 \right\} = 3a_{-2} t \quad (34.32)$$

Comparison of the latter two equations yields:

$$y_0 = \sqrt{3a_{-2}\tau_0} \quad (34.33)$$

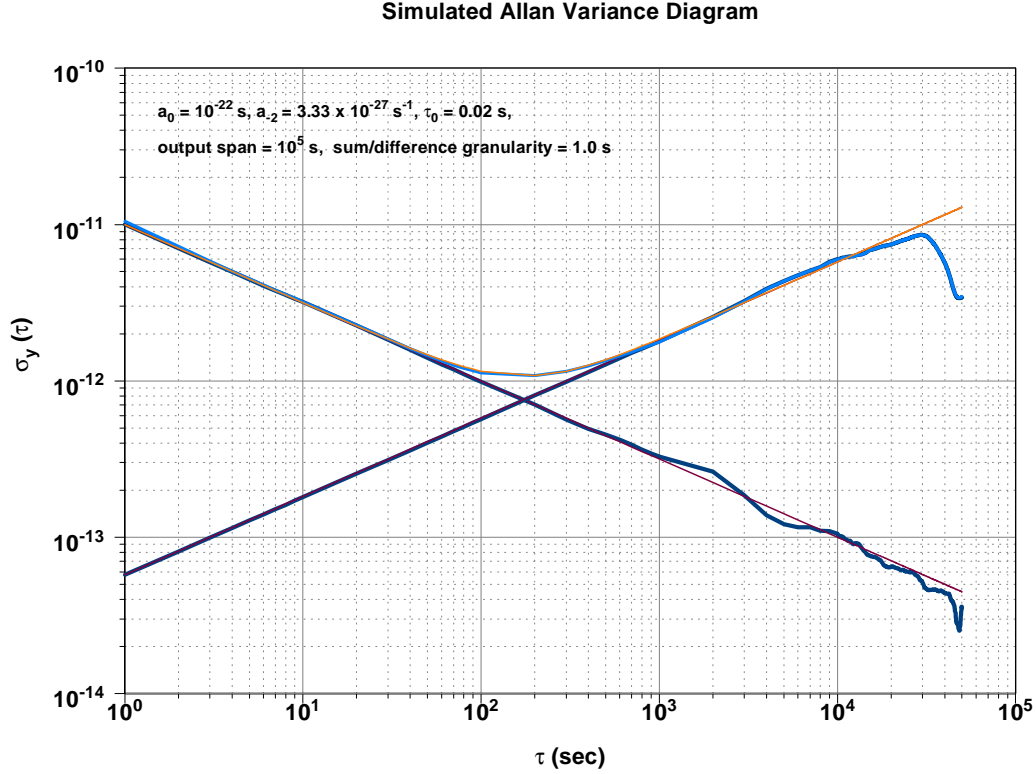


Figure 34.1: Simulated Allan Variance Diagram

Given $t_0 = 0$, and values for a_{-2} and τ_0 , derive y_0 from Eq. 34.33, and with the help of a uniform random number generator, simulate the random walk sequence in clock frequency $y(t)$, defined by Eq. 34.29, due to FM random walk.

Clock phase, due to FM random walk is derived by inserting Eq. 34.29 into Eq. 34.13:

$$x(t) = y(t) \tau_0 \quad (34.34)$$

The Allan Variance Diagram presents the logarithm (base 10) of clock sample time τ (sec) on the abscissa (x axis), and the logarithm (base 10) of the square-root $\sigma_y(\tau)$ of the Allan Variance $\sigma_y^2(\tau)$ on the ordinate (y axis). That is, $\log_{10} \sigma_y(\tau)$ is graphed as a function of $\log_{10} \tau$.

Fig. 34.1 presents an ideal simulated Allan Variance Diagram. Abscissa values $\{0,1,2,3,4,5\}$ have been replaced with $\{10^0,10^1,10^2,10^3,10^4,10^5\}$. Ordinate values $\{-10,-11,-12,-13,-14\}$ have been replaced with $\{10^{-10},10^{-11},10^{-12},$

$10^{-13}, 10^{-14}$ }. The Allan root-variance graph approximation from Fig. 34.1 is defined by the sum of the two straight lines. Frequency white noise (for a_0) is associated with the straight line with negative slope ($-1/2$). Frequency random walk (for a_{-2}) is associated with the straight line with positive slope ($1/2$). The wiggly overlaid functions are reconstructions from simulated data – they become poorer approximations as τ increases because the sample ensemble is exhausted (becomes smaller) as τ increases.

34.5.3 Filter Time Update for Clock

Let t_k and $t_{k+1} > t_k$ define the time update interval $[t_k, t_{k+1}]$. Let us denote the clock filter transition function $\Phi_{k+1,k}$, and the clock filter process noise covariance $Q_{k+1,k}$.

State Estimate

State estimate propagation for the clock:

$$X_{k+1|k} = \Phi_{k+1,k} X_{k|k} \quad (34.35)$$

where:

$$\Phi_{k+1,k} = \begin{bmatrix} 1 & (t_{k+1} - t_k) \\ 0 & 1 \end{bmatrix} \quad (34.36)$$

Error Covariance

Error covariance propagation for the clock:

$$P_{k+1|k} = \Phi_{k+1,k} P_{k|k} \Phi_{k+1,k}^T + Q_{k+1,k} \quad (34.37)$$

where:

$$Q_{k+1,k} = \begin{bmatrix} \left[a_0 (t_{k+1} - t_k) + a_{-2} (t_{k+1} - t_k)^3 \right] & \left[(a_0 + 3a_{-2} (t_{k+1} - t_k)^2) / 2 \right] \\ \left[(a_0 + 3a_{-2} (t_{k+1} - t_k)^2) / 2 \right] & \left[(a_0 / (t_{k+1} - t_k)) + 3a_{-2} (t_{k+1} - t_k) \right] \end{bmatrix} \quad (34.38)$$

$$P_{j|k} = \begin{bmatrix} P_{j|k}^x & P_{j|k}^{xy} \\ P_{j|k}^{yx} & P_{j|k}^y \end{bmatrix}, \quad j \in \{k, k+1\} \quad (34.39)$$

34.5.4 State Estimate Parameters

The fractional frequency $y(t)$ and its generalized integral $x(t)$ is estimated for the clock.

34.6 Zucca-Tavella Clock Model

Following ZT [105] Eq. (19), define

$$\bar{Y} = \frac{1}{\tau} \int_{t_k}^{t_{k+1}} y(t) dt \quad (34.40)$$

Let $E\{\cdot\}$ denote the expectation operator, and let $\sigma_y^2(\tau)$ denote the Allan Variance, defined by (ZT [105] Eq. (18))

$$\sigma_y^2(\tau) = \frac{1}{2} E \left\{ (\bar{Y}_{k+1} - \bar{Y}_k)^2 \right\} \quad (34.41)$$

Let $X_1(t)$ denote clock phase deviation, let $X_2(t)$ denote the random walk component of clock frequency deviation, and in part, let $X_3(t)$ denote clock frequency drift (aging). Then

$$X(t) = \begin{bmatrix} X_1(t) \\ X_2(t) \\ X_3(t) \end{bmatrix} \quad (34.42)$$

Denote initial conditions with

$$X(0) = \begin{bmatrix} X_1(0) \\ X_2(0) \\ X_3(0) \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} \quad (34.43)$$

Let μ_1 denote the constant initial frequency offset, let a denote deterministic constant frequency drift, let $\mu_2 = a - c_3$ denote the constant deterministic frequency aging, and let μ_3 denote the linear coefficient of the time variation of the frequency drift a . The ZT three-state clock model is defined by the stochastic differential equation (ZT [105] Eq. (1))

$$\begin{bmatrix} dX_1(t) \\ dX_2(t) \\ dX_3(t) \end{bmatrix} = \begin{bmatrix} [X_2(t) + \mu_1] dt + \sigma_1 dW_1(t) \\ [X_3(t) + \mu_2] dt + \sigma_2 dW_2(t) \\ \mu_3 dt + \sigma_3 dW_3(t) \end{bmatrix} \quad (34.44)$$

where $t \geq 0$ and $W_1(t)$, $W_2(t)$, and $W_3(t)$ are three independent one-dimensional standard Wiener processes. The closed form solution is given by ZT ([105] Equation 5, page 290)

$$X(t) = \Phi(t, t_0) X(t_0) + G(t) \quad (34.45)$$

where

$$\Phi(t, t_0) = \begin{bmatrix} 1 & t - t_0 & (t - t_0)^2 / 2 \\ 0 & 1 & t - t_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (34.46)$$

$$G(t) = \begin{bmatrix} \sigma_1 W_1(t) + \sigma_2 \int_0^t (t-s) dW_2(t) + \sigma_3 \int_0^t \left((t-s)^2 / 2 \right) dW_3(s) \\ \sigma_2 W_2(t) + \sigma_3 \int_0^t (t-s) dW_3(s) \\ \sigma_3 W_3(t) \end{bmatrix} \quad (34.47)$$

$$B(t) = \begin{bmatrix} t & t^2/2 & t^3/6 \\ 0 & t & t^2/2 \\ 0 & 0 & t \end{bmatrix} \quad (34.48)$$

The error covariance matrix associated with $X(t)$ was presented in Equation 34.6 (given initially by ZT ([105] Equation (9) page 290))

$$Q_{k+1,k} = \begin{bmatrix} \sigma_1^2 \tau + \sigma_2^2 \frac{\tau^3}{3} + \sigma_3^2 \frac{\tau^5}{20} & \sigma_2^2 \frac{\tau^2}{2} + \sigma_3^2 \frac{\tau^4}{8} & \sigma_3^2 \frac{\tau^3}{6} \\ \sigma_2^2 \frac{\tau^2}{2} + \sigma_3^2 \frac{\tau^4}{8} & \sigma_2^2 \tau + \sigma_3^2 \frac{\tau^3}{3} & \sigma_3^2 \frac{\tau^2}{2} \\ \sigma_3^2 \frac{\tau^3}{6} & \sigma_3^2 \frac{\tau^2}{2} & \sigma_3^2 \tau \end{bmatrix} \quad (34.49)$$

ZT derived the Allan Variance $\sigma_y^2(t_k, \tau)$ of Equation 34.50 (or ZT [105] Equation 35) from the Equations above to find that the Allan Variance is a function of t_k as well as τ

$$\sigma_y^2(t_k, \tau) = \left[\frac{\sigma_1^2}{\tau} + \frac{\sigma_2^2 \tau}{3} + \frac{\sigma_3^2 \tau^3}{20} \right] + \sigma_3^2 \left[\frac{\tau^3}{3} + \frac{\tau^2 t_k}{2} \right] + \frac{\tau^2}{2} [c_3 + \mu_3 [\tau + t_k]]^2 \quad (34.50)$$

The estimation of $X_3(t)$ absorbs μ_3 and c_3 . In this sense μ_3 and c_3 are ignorable for filter-smoother estimation, and Equation 34.50 becomes

$$\sigma_y^2(t_k, \tau) = \frac{\sigma_1^2}{\tau} + \frac{\sigma_2^2 \tau}{3} + \frac{23\sigma_3^2 \tau^3}{60} + \frac{\sigma_3^2 \tau^2 t_k}{2} \quad (34.51)$$

But $\sigma_y^2(t_k, \tau)$, according to Equation 34.51, goes infinite with increasing $t_k > 0$. ZT have demonstrated that $\sigma_y^2(t_k, \tau)$ is a useless measure of clock C_i fractional frequency stability when $\sigma_3 > 0$. In this case one can use the Hadamard variance, defined in ZT Equation (44) and demonstrated to be free of t_k in ZT Equation (45), for assessment of clock C_i fractional frequency stability.

The ODTK filter-smoother estimates the 3×1 matrix clock state $X(t)$ for each clock C_i according to Equations 34.42 through 34.49, where $Q_{k+1,k}$ is used for clock C_i filter process noise covariance, and the estimation of $X_3(t)$ (aging) is user optional. Allan parameter values for a_0 , a_{-2} , and ZT diffusion coefficient σ_3 are input to the filter by the user. Allan parameter values for a_0 and a_{-2} can be determined by the user by inspection of user Allan variance diagram for each clock C_i . The mapping between ZT diffusion coefficients, Allan parameters, and GEONS parameters is presented in Table 34.1. The GEONS[32] deterministic relativistic clock phase drift Δd_{Rel}^m for each satellite clock is modeled in ODTK with user option selection.

Part VIII

Least Squares (LS)

Chapter 35

LS Inputs

35.1 Input Values

Let y_j denote a $K \times 1$ column matrix of new measurement values y_j^i , with K measurement types $i \in \{1, 2, \dots, K\}$, at measurement time t_j , where $j \in \{1, 2, \dots, M\}$. Thus:

$$y_j = \begin{bmatrix} y_j^1 \\ y_j^2 \\ \vdots \\ y_j^K \end{bmatrix} \quad (35.1)$$

Let y denote a $KM \times 1$ column matrix with matrix elements y_j :

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix} \quad (35.2)$$

Let W_j denote a $K \times K$ diagonal matrix of measurement weights at time t_j , where again, superscripts denote measurement types:

$$W_j = \begin{bmatrix} W_j^1 & & & 0 \\ & W_j^2 & & \\ & & \ddots & \\ 0 & & & W_j^K \end{bmatrix} \quad (35.3)$$

where:

$$W_j^i = \frac{1}{(\sigma_j^i)^2} \quad (35.4)$$

and where σ_j^i is the error sigma on measurement type i at time t_j . Let W denote a $KM \times KM$ diagonal matrix with diagonal matrix elements W_j :

$$W = \begin{bmatrix} W_1 & & & 0 \\ & W_2 & & \\ & & \ddots & \\ 0 & & & W_M \end{bmatrix} \quad (35.5)$$

Let $\hat{X}_{0|m}$ denote the a priori $n \times 1$ matrix state estimate at epoch time t_0 , where m denotes the time t_m of last measurement used, and the state estimate size is $n \geq 6$. If $t_0 = t_m$, then $\hat{X}_{0|m} = \hat{X}_{m|m} = \hat{X}_{0|0}$. Since one can always make this happen with a state estimate propagation there is no loss of generality in assuming it is true. The state estimate $\hat{X}_{0|0}$ always contains a 6×1 matrix orbit estimate $\hat{Z}_{0|0}$.

We are given numerical values for the initial state estimate matrix $\hat{X}_{0|0}$, measurement matrix y , and measurement weight matrices W . Note: It is operationally conventional to specify W_j as a *constant* for all t_j .

35.2 Initial Calculated Values

35.2.1 Measurement Residuals

Let $y^i(\hat{X}_{j|0})$ denote the nonlinear scalar measurement representation for measurement type i at measurement time t_j . Let $y(\hat{X}_{j|0})$ denote a $K \times 1$ column matrix of measurement representations $y^i(\hat{X}_{j|0})$ at time t_j :

$$y(\hat{X}_{j|0}) = \begin{bmatrix} y^1(\hat{X}_{j|0}) \\ y^2(\hat{X}_{j|0}) \\ \vdots \\ y^K(\hat{X}_{j|0}) \end{bmatrix} \quad (35.6)$$

Let $\Delta y(\hat{X}_{j|0})$ denote the $K \times 1$ column matrix of measurement residuals for time t_j defined by:

$$\Delta y(\hat{X}_{j|0}) = y_j - y(\hat{X}_{j|0}) \quad (35.7)$$

and adopt the notation:

$$\Delta y_j = \Delta y(\hat{X}_{j|0}) \quad (35.8)$$

Let Δy denote the $KM \times 1$ matrix with matrix elements Δy_j :

$$\Delta y = \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \\ \vdots \\ \Delta y_M \end{bmatrix} \quad (35.9)$$

35.2.2 Partial Derivatives

Let H_j^i denote the $1 \times n$ measurement-state Jacobian of partial derivatives for time t_j and measurement type i :

$$H_j^i = \frac{\partial y_j^i}{\partial X_j} \quad (35.10)$$

Let $\Phi_{j,0}$ denote the $n \times n$ state-transition matrix Jacobian of partial derivatives for times t_0 and t_j :

$$\Phi_{j,0} = \frac{\partial X_j}{\partial X_0} \quad (35.11)$$

Define the $1 \times n$ matrix A_j^i :

$$A_j^i = H_j^i \Phi_{j,0} \quad (35.12)$$

Form the $K \times n$ matrix A_j :

$$A_j = \begin{bmatrix} A_j^1 \\ A_j^2 \\ \vdots \\ A_j^K \end{bmatrix} \quad (35.13)$$

Form the $KM \times n$ matrix A :

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_M \end{bmatrix} \quad (35.14)$$

Chapter 36

Least Squares Solutions

36.1 LS Normal Equation and Solution

Let X_0 denote the true unknown state, let $y(X_0)$ denote the unknown measurement representation of y as a function of X_0 , and define a performance function J :

$$J = (y - y(X_0))^T W (y - y(X_0)). \quad (36.1)$$

Expand $y(X_0)$ about the known state estimate $\hat{X}_{0|0}$ in a multi-dimensional Taylor's series:

$$y(X_0) = y(\hat{X}_{0|0}) + \left[\frac{\partial y(X)}{\partial X} \right]_{\hat{X}_{0|0}} (X_0 - \hat{X}_{0|0}) + \{\text{higher order terms}\} \quad (36.2)$$

Define:

$$A = \left[\frac{\partial y(X)}{\partial X} \right]_{\hat{X}_{0|0}} \quad (36.3)$$

$$\Delta X_0 = [X_0 - \hat{X}_{0|0}] \quad (36.4)$$

36.1.1 Linearization

Insert Eq. 36.2 into Eq. 36.1, ignoring higher order terms, to get:

$$J = \left([y - y(\hat{X}_{0|0})] - A\Delta X_0 \right)^T W \left([y - y(\hat{X}_{0|0})] - A\Delta X_0 \right) \quad (36.5)$$

Define:

$$\Delta y = [y - y(\hat{X}_{0|0})] \quad (36.6)$$

to write:

$$J(\Delta X_0) = (\Delta y - A\Delta X_0)^T W (\Delta y - A\Delta X_0). \quad (36.7)$$

Differentiate J with respect to ΔX_0 :

$$\frac{\partial J}{\partial \Delta X_0} = 2(A^T W A \Delta X_0 - A^T W \Delta y) \quad (36.8)$$

and set the gradient $\partial J / \partial \Delta X_0$ equal to zero to find extrema. This yields:

$$A^T W A \Delta \hat{X}_{0|M} = A^T W \Delta y, \quad (36.9)$$

where the use of $\Delta \hat{X}_{0|M}$ denotes the value of ΔX_0 that extremizes $J(\Delta X_0)$. See Appendix A for details on calculation of the derivative $\partial \tilde{J} / \partial \Delta X_0$. Eq. 36.9 is called the least squares *normal equation*. If $A^T W A$ is non-singular, the least squares solution is obtained by solving Eq. 36.9 for $\Delta \hat{X}_{0|M}$, and adding $\Delta \hat{X}_{0|M}$ to $\hat{X}_{0|0}$. Thus:

$$\Delta \hat{X}_{0|M} = (A^T W A)^{-1} (A^T W \Delta y) \quad (36.10)$$

$$\hat{X}_{0|M} = \hat{X}_{0|0} + \Delta \hat{X}_{0|M} \quad (36.11)$$

36.2 Remove the Squaring Operation

Let $W^{1/2}$ denote the square root matrix to the diagonal matrix W . That is: $W = W^{1/2} W^{1/2}$. Define $B = W^{1/2} A$, and $b = W^{1/2} \Delta y$. Then Eq. 36.9 can be written:

$$(B^T B) \Delta \hat{X}_{0|M} = B^T b \quad (36.12)$$

Consider the equation:

$$B \Delta \hat{X}_{0|M} = b \quad (36.13)$$

Notice that Eq. 36.12 is rigorously derived from Eq. 36.13 by multiplication with B^T . These equations are theoretically equivalent. But given a fixed length computer mantissa, they are not numerically equivalent. This is because the squaring operation $B^T B$ halves the significance of $\Delta \hat{X}_{0|M}$ ($= (B^T B)^{-1} B^T b$) relative to the fixed length computer mantissa. A solution of Eq. 36.13 directly for $\Delta \hat{X}_{0|M}$ can recover the lost numerical significance.

36.3 Solution by Triangularization of B

The least squares equation:

$$B \Delta \hat{X}_{0|M} = b \quad (36.14)$$

must be solved for $\Delta \hat{X}_{0|M}$, where B is a $KM \times n$ matrix with rank n , $KM \geq n$, and b is an $KM \times 1$ matrix. The solution seems trivial when $KM = n$:

$$\Delta \hat{X}_{0|M} = B^{-1}b \quad (36.15)$$

For $KM > n$, the solution of Eq. 36.14 with greatest numerical stability requires the use of orthogonal Householder transformations (Lawson & Hanson [66], page 121) for triangularization of matrix B . Our problem is to calculate an acceptable solution of Eq. 36.14 when $KM > n$.

36.3.1 Solution Overview

A $KM \times KM$ orthogonal matrix T is calculated such that the upper $n \times n$ matrix of the $KM \times n$ matrix TB is upper triangular. Then:

$$TB \Delta \hat{X}_{0|M} = Tb \quad (36.16)$$

is easily solved for $\Delta \hat{X}_{0|M}$ with back substitutions because TB is upper triangular.

Chapter 37

Least Squares Inadequacies

37.1 LS Measurement Residuals

A fundamental hypothesis for the least squares algorithm says that measurement residuals are white noise. This is desirable because white noise residuals demonstrate the extraction of all information from the measurements. But anyone performing orbit determination who has examined least squares measurement residuals due to processing *real data* knows that they are serially correlated, not white. This glaring contradiction demonstrates the inadequacy of the least squares algorithm for performing orbit determination.

37.2 Incomplete LS Model

The LS model equation has no structure to account for serially correlated random gravity modeling errors, serially correlated random air-drag modeling errors, and serially correlated random solar pressure modeling errors. These errors are absent in the LS state error transition function. The least squares state estimate structure is necessarily incomplete because least squares cannot accommodate random state errors (Bucy and Joseph [10], page 140); *e.g.*, the LS estimated drag coefficient and range bias parameters are necessarily modeled as *time constants*.

37.3 Least Squares State Error Covariance

The LS state error covariance matrix $(A^TWA)^{-1}$ is a function only of the measurement error covariance matrix W^{-1} . There is no structure here for force modeling errors.

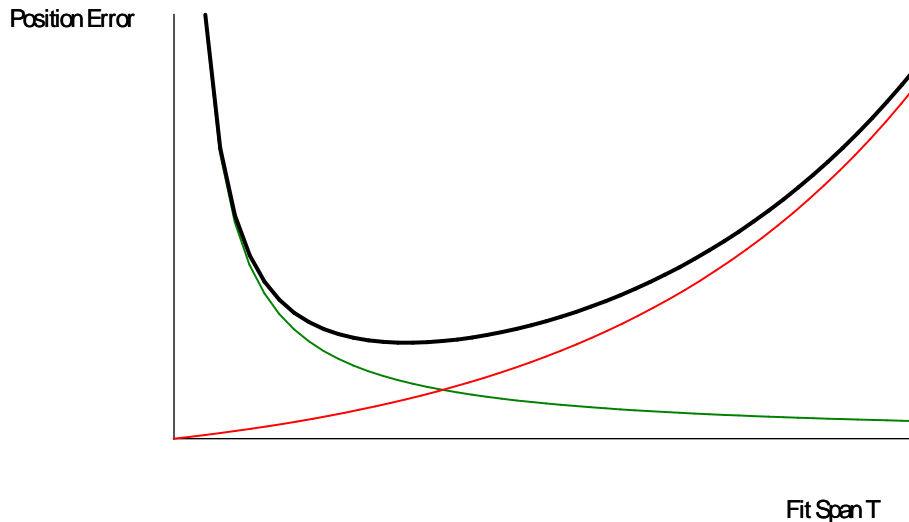


Figure 37.1: Position Error vs Least Squares Fit Span

37.4 Batch Simultaneity

Physically, the alternating effects of force modeling errors on the state estimate and the collection of measurement information for the state estimate evolves sequentially with time. But the least squares algorithm maps measurement information at distinct times *simultaneously* to the state estimate. This provides no *time-sequential* structure to accommodate force modeling errors.

37.5 Schmidt's Analysis of Least Squares

Figure 37.1 illustrates Stan Schmidt's[100] analysis of the ephemeris predicted position error magnitude in one position component (*e.g.*, intrack), at an arbitrarily selected fixed prediction time, due to variations in the least squares fit span T (units time). The green colored function (always decreasing with LS fit span) is due to measurement error, the red colored function (always increasing with LS fit span) is due to force model error, and the heavy black function (superposition) is due to the sum of measurement error and force model error.

37.5.1 A Practical Experiment

Figure 37.1 has practical utility. A sample superposition function can be constructed using real data. Let t_{BD} and t_{ED} denote the times of beginning and end of the tracking data span, let t_{ED} be the fixed least squares epoch, and let

t_P denote a fixed prediction time, where $t_{BD} < t_{ED} < t_P$. With t_{ED} fixed the propagation time $t_P - t_{ED}$ is also fixed. The fit span $T = t_{ED} - t_{BD}$ is variable, and thus t_{BD} is variable.

First perform a least squares fit with conjectured fit span T_C centered on fixed time t_P , and save the resulting ephemeris across time t_P . Call this the *reference* ephemeris. Then perform a sequence $j \in \{1, 2, 3, \dots\}$ of least squares fits where each fit span T_j is distinct, but terminates at the fixed time t_{ED} . For each fit propagate an ephemeris across t_P , and difference the propagated ephemeris with the reference ephemeris at time t_P . Graph the magnitude of this difference as a function of T_j to form the superposition from measurement errors and force modeling errors. If variations in T_j are sufficiently small and sufficiently large, the implied superposition function will have negative slope to the left and positive slope to the right with a unique minimum. Let T_{BEST} denote the fit span associated with the superposition minimum. Now replace T_C with T_{BEST} and repeat the experiment. Operationally, one must identify the superposition *minimum* in order to "optimize" least squares performance for operational use. The final reference ephemeris will have errors, but their magnitudes will be small, relative to the errors in the ephemerides used to determine the "optimal" fit span, if $t_P - t_{ED}$ is sufficiently long.

All three functions (red, green, and black) can be graphed quantitatively using realistic simulated tracking data.

An Unwanted Surprise

Least squares orbit determination has the surprising property that when one adds more information (after the resultant superposition minimum) the LS accuracy performance gets worse! Generally, LS accuracy performance depends on both data span and data density with time. Performance will also suffer due to non-uniform data density with time! Recall Eqs. 36.7, 36.8, and 36.9: Least squares chooses a state estimate that minimizes the sum of squares of weighted tracking data residuals. It would be preferable to minimize state estimate error variances.

37.6 Gibb's Effect

Our presentation of the *Gibbs' effect*¹ is mentioned here so as to distinguish it from Stan Schmidt's analysis. Consider fitting any engineering data with uniform density and uniform weight to any appropriate smooth approximating function using a least squares curve fit, where the number of data points is greater than the number of degrees of freedom in the smooth function.

Willard Gibbs noticed that the expected curve fit error magnitudes inside the data span are always greatest near both ends of the data span. That is, data neighborhoods containing the points of initiation and termination of data

¹An appropriate discussion of the *Gibbs Effect* relating to least squares estimation is presented in a paper by Foster and Richards[25].

produce degraded estimates as compared to data neighborhoods interior to the data span. The best least squares estimate is at the center of the data span. The least squares signed error envelope can be visualized as a *bow tie*. The *Gibbs' effect* contains no notion of force model, and no notion of an arbitrarily selected fixed prediction point.

This is useful for least squares error analysis, but should not be confused with Stan Schmidt's analysis. Schmidt refers to a fixed prediction time for least squares orbit determination, and compares the effects of measurement error and force model error due to data span *length* at that fixed prediction time.

Chapter 38

Tracking Data Editing

One of the most difficult problems in the use of iterated least squares corrections for operational orbit determination derives from outliers in the tracking data. Thus tracking data editing is particularly a least squares related problem. Any least squares measurement residual *variance* is useless for tracking data editing because the state estimate error variance contribution is always significantly too small – due to the absence of a mechanism to represent spacecraft force modeling error covariance.

Refined least squares tracking data editing is typified by an algorithm to identify measurement residual magnitudes that are large with respect to the root mean square (RMS) on the total collection of measurement residuals used in the least squares batch. This approach is effective when most of the data are *not* outliers and when the a priori orbit estimate error magnitudes are sufficiently small to enable iterative reduction in the residual RMS. Iterated least squares corrections are intended to cope simultaneously with nonlinearities (*i.e.*, large error magnitudes in the a priori orbit estimate) and with tracking data editing. When the iterative relative RMS ratio magnitude is reduced to below an a priori defined epsilon, then the iteration is terminated – some say converged. But when tracking data are removed from the batch, the residual RMS is redefined together with the effective editing policy. Thus the effective editing policy is redefined on each iteration. It is thus not surprising that operational least squares methods employ a capability to terminate the iterative corrections arbitrarily after an a priori defined maximum iteration count.

Democracy rules: *e.g.*, Given a batch of measurements where seventy per cent of the data are biased outliers, the RMS residual editor will throw out the good data to derive a least squares fit to the outliers.

We shall use the iterative least squares RMS editing method for least squares orbit determination, but only after the application of every other effective data editor. We should note that our autonomous optimal filter editor should be used in preference to the iterative least squares RMS editor. Unfortunately, we only use the least squares estimate as a bridge between IOD and filter initialization.

38.1 Tracking Data Editor Identification

Editors are applied to each station pass. Let m denote the number of data sets initially given in the station pass, each at a distinct time, say t_k , $k \in \{1, 2, 3, \dots, m\}$. Each editor will potentially reduce the value of m as outliers are discarded. In order to complete the station pass editing, a minimum number of measurement sets must survive initial editors.

In order to support a successful least squares correction the measurements must span a particular orbit-period dependent time span. For otherwise, the orbit estimate will be unobservable to the batch of measurements used.

The editors identified below are to be applied in the order given.

1. Minimum Number of Data Sets Editor
2. Minimum Time Span Editor
3. Tracking Loop Flag Editor
4. Repeat: Minimum Number of Data Sets Editor
5. Repeat: Minimum Time Span Editor
6. Gross Raw Data Editor
7. Repeat: Minimum Number of Data Sets Editor
8. Repeat: Minimum Time Span Editor
9. Minimum Elevation Editor
10. Repeat: Minimum Number of Data Sets Editor
11. Repeat: Minimum Time Span Editor
12. Sliding Raw Data Polynomial Editor
13. Repeat: Minimum Number of Data Sets Editor
14. Repeat: Minimum Time Span Editor
15. IOD Sliding Fit Kepler Element Editor
16. Repeat: Minimum Time Span Editor
17. Second Difference Editor Using IOD
18. Repeat: Minimum Time Span Editor
19. Residual RMS Editor Criterion Using IOD
20. Repeat: Minimum Time Span Editor
21. Least Squares Solution Attempt

- 22. Iterative RMS convergence
- 23. Exceed Maximum Iteration Count Editor
- 24. Manual Editor
- 25. Select A New Station Pass
- 26. Measurement Bias Estimate/Guess
- 27. Call an Orbit Analyst
- 28. Call AGI

38.2 Minimum Number of Data Sets Editor

Let T_m denote the minimum number of measurement data sets required per station pass. Let m denote the number of data sets surviving for the station pass, each at a distinct time, say t_k , where $k \in \{1, 2, 3, \dots, m\}$.

If $m < T_m$, (default 3) then discard the given station pass, select another, and begin editing again.

38.3 Minimum Time Span Editor

With some measurement sets the least squares orbit solution is not observable without measurements that span a time interval defined by multiple orbit periods. Provide a user defined constant T_{MT} for minimum time span associated with each measurement type and assumed orbit class.

Let $[t_B, t_E]$ denote the measurement batch time-span currently available. If $|t_E - t_B| < T_{MT}$, acquire more tracking data so as to satisfy the requirement: $|t_E - t_B| \geq T_{MT}$.

Note that multiple station passes will probably be required to satisfy this requirement.

38.4 Tracking Loop Flag Editor

If the measurement file includes a tracking loop flag that is set FALSE by sensor hardware, then discard the measurement.

38.5 Gross Raw Data Editor

Let η denote a measurement value of a particular type from a particular measurement set. An orbit class, tracking system, and scale factor value $S_\eta > 1$ are selected by the user for input to the gross editor algorithm. The gross editor algorithm uses these inputs to define a gross threshold value $T_\eta > 0$ for each

measurement type η . Let $\bar{\eta}_f > 0$ denote the class dependent, tracking system dependent, and measurement type dependent upper bound for the error-free component of η . The white noise and bias on η are negligible as compared to $\bar{\eta}_f$. Then T_η is defined by:

$$T_\eta = S_\eta \bar{\eta}_f$$

Edit Test: Compare each measurement value magnitude $|\eta|$ with T_η . If $|\eta| < T_\eta$, accept measurement $\eta(t)$. Otherwise discard $\eta(t)$ and the entire measurement set associated with $\eta(t)$ at time t .

Example: Let $\eta(t) = \rho(t)$ denote a range measurement for the SCF tracking system and a LEO spacecraft. Then $\rho(t)$ is a two-way range. If the orbit class is LEO and the spacecraft position vector length $r_{LEO} < 1.3er$, then $\bar{\rho}_f = 2\sqrt{1.3^2 - 1^2}er = 1.66er$. If $S_\eta = 1.1$, then $T_\rho = 1.1 \times 1.66er = 1.826er$. Edit test: Given a range measurement $\rho(t)$, if $|\rho(t)| < 1.826er$, accept range value $\rho(t)$ for processing, otherwise discard $\rho(t)$ and associated measurements at time t .

Redefine m to denote the number of surviving data sets.

38.6 Minimum Elevation Editor

If elevation values E_j are given in the raw data, then reject all surviving measurement sets for which $E_j < T_{EL}$, where

T_{EL} is a user defined quantity. Default: $T_{EL} = 15$ deg.

38.7 Sliding Raw Data Polynomial Editor

Provide user defined constants for polynomial order O_P , minimum number of measurements N_m , and RMS epsilon $\epsilon_{RMSpoly}$. Perform a linear least squares fit of an N^{th} order polynomial to an overlapping sliding batch of N_m measurements for each measurement type. Calculate a polynomial residual Δy_j RMS for each measurement of each fit, then discard each measurement whose residual magnitude exceeds $\epsilon_{RMSpoly}$. See definitions for the polynomial least squares method in the IOD section herein.

38.8 Kepler Element Editor

Perform an IOD sliding fit to all m surviving measurement sets, with each set associated with a time, say t_k , $k \in \{1, 2, 3, \dots, m\}$.

Example: If IOD is Herrick-Gibbs, then three measurement sets of range, azimuth, and elevation are required, say at times t_j , t_{j+1} , and t_{j+2} , for each solution. The first Herrick-Gibbs solution for position and velocity components at time t_2 use measurement sets at times t_1 , t_2 , and t_3 . The second Herrick-Gibbs solution for position and velocity components at time t_3 use measurement

sets at times t_2 , t_3 , and t_4 . The j^{th} Herrick-Gibbs solution for position and velocity components at time t_{j+1} use measurement sets at times t_j , t_{j+1} , and t_{j+2} . And the m^{th} Herrick-Gibbs solution for position and velocity components at time t_{m-1} use measurement sets at times t_{m-2} , t_{m-1} , and t_m .

Transform each solution for position and velocity to Kepler elements.

38.8.1 Kepler Element Bounds Criterion

Check each IOD solution to see that each Kepler element within each solution is within its formal bound. The following orbit bound definitions values should be made consistent with the existing orbit class definitions.

LEO

$$\left[\begin{array}{l} 1.01er \leq a < 1.30er \\ 0 \leq e < 0.1 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 0 \text{ deg} \leq i \leq 180 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

MEO

$$\left[\begin{array}{l} 1.3er \leq a < 4.0er \\ 0 \leq e \leq 0.1 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 0 \text{ deg} \leq i \leq 180 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

GPS

$$\left[\begin{array}{l} 4.0er \leq a < 4.4er \\ 0 \leq e \leq 0.1 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 53 \text{ deg} \leq i \leq 57 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

SUB-GEO

$$\left[\begin{array}{l} 4.0er \leq a < 6.4er \\ 0 \leq e \leq 0.1 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 0 \text{ deg} \leq i \leq 180 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

GEO

$$\left[\begin{array}{l} 6.4er \leq a \leq 6.8er \\ 0 \leq e \leq 0.1 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 0 \text{ deg} \leq i \leq 180 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

SUPER-GEO

$$\left[\begin{array}{l} 6.8er \leq a \leq 1000er \\ 0 \leq e \leq 0.1 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 0 \text{ deg} \leq i \leq 180 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

MOLNIYA

$$\left[\begin{array}{l} 4.0er \leq a \leq 4.4er \\ 0.65 \leq e \leq 0.75 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 62 \text{ deg} \leq i \leq 66 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 260 \text{ deg} \leq \omega < 280 \text{ deg} \end{array} \right]$$

HEO

$$\left[\begin{array}{l} 1.3er \leq a \leq 5.0er \\ 0.1 \leq e \leq 0.999 \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 0 \text{ deg} \leq i \leq 180 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

OTHER

$$\left[\begin{array}{l} ?er \leq a \leq ?er \\ ? \leq e \leq ? \\ 0 \text{ deg} \leq u < 360 \text{ deg} \\ 0 \text{ deg} \leq i \leq 180 \text{ deg} \\ 0 \text{ deg} \leq \Omega < 360 \text{ deg} \\ 0 \text{ deg} \leq \omega < 360 \text{ deg} \end{array} \right]$$

Measurement Set Selection

Continuing with the Herrick-Gibbs example, Let $m \geq 3$ denote a given number of measurement sets within the same tracking station pass, where each set consists of range, azimuth, and elevation measurements, and each set is associated with a distinct time tag. Each execution of the Herrick-Gibbs method requires the input of three distinct sets of range, azimuth, elevation, and time tag. Transform each solution from position and velocity components to Kepler elements.

38.8.2 $m = M = 3$

Calculate one Herrick-Gibbs solution for the Kepler element bounds check.

38.8.3 $m = M = 4$

With $m = 4$, we can calculate exactly 4 distinct Herrick-Gibbs solutions:

$$\binom{4}{3} = \frac{4!}{3!(4-3)!} = 4$$

Notice that all four solutions are correlated because at least one of the measurement sets used in each solution is also contained in each of the other three solutions. Name the four measurement sets a, b, c, d , and identify the four solutions with:

$$\begin{bmatrix} S_1 & abc \\ S_2 & bcd \\ S_3 & abd \\ S_4 & acd \end{bmatrix}$$

Each measurement set occurs in exactly three solutions. It is logically impossible that three solutions could succeed and one solution could fail, or that two solutions could succeed and two solutions could fail. For if two solutions succeed, then measurement sets a, b, c, d are good.

Given that all four solutions succeed, then all four measurement sets are accepted. If exactly three solutions succeed and one solution fails, then reject all four measurement sets. If exactly two solutions succeed and two fail, then reject all four measurement sets.

Given that one solution succeeds but the other three fail, then three of the four measurement sets are accepted and one set is discarded. Example: Suppose no bound is violated for solution S_1 , but violations are incurred for solutions S_2, S_3 , and S_4 . Then measurement sets a, b , and c are accepted, and d is discarded.

38.8.4 $m = M = 5$

With $m = 5$, we can calculate exactly 10 distinct Herrick-Gibbs solutions:

$$\binom{5}{3} = \frac{5!}{3!(5-3)!} = 10$$

Name the five measurement sets a, b, c, d, e , and identify the ten solutions with:

$$\left[\begin{array}{l} S_1 \quad abc \\ S_2 \quad abd \\ S_3 \quad abe \\ S_4 \quad acd \\ S_5 \quad ace \\ S_6 \quad ade \\ S_7 \quad bcd \\ S_8 \quad bce \\ S_9 \quad bde \\ S_{10} \quad cde \end{array} \right]$$

Each measurement set occurs in exactly six solutions. Given that all ten solutions succeed, then all ten measurement sets are accepted. Given that four solutions succeed and six solutions fail, then the one measurement set common to the six failures is discarded, and the other four measurement sets are accepted. Given that one solution succeeds and nine fail, then accept the three measurement sets associated with the successful solution and reject the other two measurement sets. For any other combination of success and failures, reject all five measurement sets.

38.8.5 $m \geq 6$

Process all measurement sets in an overlapping sliding group of five, using the last procedure for $M = 5$.

Note: We could generalize the combinatorials for any finite integer m , but this might be an unusual expense in programming and CPU.

38.8.6 Kepler Element Statistics Criterion

Provide user inputs for constant thresholds T_a , T_e , and T_i , with default values set to 3. Consider the measurement sets that survive the Kepler Element Bounds Criterion. For each station pass calculate the sample mean μ_β and sample variance σ_β^2 about the mean for Kepler orbit elements $\beta \in \{a, e, i\}$. Compare each element value difference magnitude $|\beta - \mu_\beta|$ with the product of its threshold with σ_β . If $|\beta - \mu_\beta| < T_\beta \sigma_\beta$, then retain the measurement set associated with β , where $\beta \in \{a, e, i\}$. Otherwise discard the measurement set associated with β . Use the same logic for identification of the measurement set to discard as used for the Kepler Element Bounds Criterion .

38.9 Second Difference Editor Using IOD

Calculate measurement residuals (first differences) using one orbit estimate from IOD. See Attachment A for second difference editing.

38.10 Convergence Criteria

38.10.1 Residual RMS Convergence Criterion Using IOD

Provide two user defined small positive constants: ϵ_{RMS} (epsilon for root mean square) and ϵ_{RRMS} (epsilon for relative root mean square).

Given measurements y_j , $j \in \{1, 2, 3, \dots, m\}$, calculate measurement residuals Δy_j by propagating one IOD orbit estimate $\hat{X}_0^{(0)}$ from t_0 to t_j to get $\hat{X}_j^{(0)}$ for each j , by calculating the measurement representation $y(\hat{X}_j^{(0)})$, and by forming the differences $\Delta y_j = y_j - y(\hat{X}_j^{(0)})$. Define:

$$A_j = \frac{\partial y_j}{\partial X_j} \frac{\partial X_j}{\partial X_0} = \frac{\partial y_j}{\partial X_j} \Phi_{j,0}$$

where theoretically:

$$\Delta X_j = \frac{\partial X_j}{\partial X_0} \Delta X_0$$

and define:

$$\Delta y = \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \\ \vdots \\ \Delta y_m \end{bmatrix}$$

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{bmatrix}$$

Then the root mean square ($RMS^{(k)}$) on the measurement residuals for iteration k , $k \in \{1, 2, \dots, L\}$, is defined with:

$$RMS^{(k)} = \frac{1}{m} (\Delta y^T \Delta y)^{1/2} = \frac{1}{\sqrt{m}} \left(\sum_{j=1}^m \Delta y_j^2 \right)^{1/2}$$

Using least squares, an orbit estimate correction $\Delta \hat{X}_0^{(k)}$ is derived for iteration k . Given the orbit estimate correction $\Delta \hat{X}_0^{(k)}$, the predicted root mean square (PRMS) for iteration $k = 1$ is defined by:

$$PRMS^{(k)} = \frac{1}{\sqrt{m}} \left((\Delta \hat{X}_0^{(k)})^T A^T A (\Delta \hat{X}_0^{(k)}) + \Delta y^T (\Delta y - 2A\Delta \hat{X}_0^{(k)}) \right)^{1/2}$$

If:

$$\frac{|PRMS^{(k)} - RMS^{(k)}|}{RMS^{(k)}} < \epsilon_{RRMS}$$

or if:

$$RMS^{(k)} < \epsilon_{RMS}$$

define convergence. Otherwise, iterate.

PRMS Derivation

The least squares solution for an iterative linear correction $\Delta \hat{X}_0$ is derived from the approximation:

$$A\Delta \hat{X}_0 \cong \Delta y$$

which has an error δy :

$$\delta y = \Delta y - A\Delta \hat{X}_0$$

Define the conditional variance on δy :

$$\sigma_{\delta y}^2 = E \left\{ (\delta y | \Delta \hat{X}_0)^T (\delta y | \Delta \hat{X}_0) \right\}$$

to get:

$$\sigma_{\delta y}^2 = \frac{1}{m} \left[(\Delta \hat{X}_0)^T A^T A (\Delta \hat{X}_0) + \Delta y^T (\Delta y - 2A\Delta \hat{X}_0) \right]$$

and assign:

$$PRMS = \sigma_{\delta y}$$

38.10.2 Residual RMS Convergence Criterion After First Correction

If:

$$\frac{|RMS^{(k+1)} - RMS^{(k)}|}{RMS^{(k)}} < \epsilon_{RRMS}$$

or if:

$$RMS^{(k+1)} < \epsilon_{RMS}$$

define convergence. Otherwise, keep iterating.

38.11 Residual Editor

Compare the magnitude of each measurement residual $|\Delta y_j|$ to the product of a scale factor N (default $N = 3$) with its unweighted *RMS*. If:

$$|\Delta y_j| < N \text{ RMS, accept the measurement}$$

$$|\Delta y_j| \geq N \text{ RMS, reject the measurement}$$

The *RMS* is used for editing prior to making each iterative least squares correction.

38.12 Least Squares Solution

Given the application of Editors (1) through (12), attempt a least squares solution with RMS residual editing.

38.12.1 Exceed Maximum Iteration Count Editor

Provide a user defined positive constant integer N_{\max} , to define maximum iteration count. If the least squares iterations exceed N_{\max} , then terminate the iterations and invoke the manual editor.

38.13 Manual Editor

Present the user with a graphical display of measurement residuals, and a capability to manually discard each associated measurement.

38.14 Select A New Station Pass

Provide a user defined positive constant integer P_{\max} . Having attempted every useful tracking data editing algorithm with subsequent failure in a least squares attempt on a particular station pass, it is time to discard the entire station pass, select a new station pass, and try again. Discontinue after P_{\max} station pass failures.

38.15 Measurement Biases

It is possible that all measurements of a particular type are significantly biased; *e.g.*, range transponder bias. If there is a resource available from which to obtain a bias estimate, try that. If not, guess a bias estimate, and try again. Finally, call an orbit analyst for help.

Part IX

Initial Orbit Determination

Chapter 39

IOD Methods

Existing validated initial orbit determination (IOD) methods are typically dependent on three measurement subsets at three distinct times, where each subset contains the same measurement types. From the measurement type set {range, range-rate, azimuth, elevation}, where azimuth and elevation never occur separately, there are two subsets of three measurement types: {range, azimuth, elevation}, {range-rate, azimuth, elevation}; there are two subsets of two measurement types: {azimuth, elevation}, {range, range-rate}; and there are two subsets of one measurement type: {range} and {range-rate}. For operational geocentric orbits, validated methods exist for {range, azimuth, elevation}, {range-rate, azimuth, elevation}, and {azimuth, elevation}. Two methods are proposed for {range} with six range values. There exist no methods for {range-rate}.

Two successfully tested methods are available in ODTK: The Herrick-Gibbs IOD method[33] for three sets of {range, azimuth, elevation}, and Robert Gooding's method[30] for three sets of {azimuth, elevation}.

39.1 Common Modeling Limitations

39.1.1 Two-Body Dynamics

In common, all IOD *problems* are six-dimensional and nonlinear, and all IOD *solutions* use two-body gravitational accelerations for orbit propagation. If the measurement error magnitudes are significant, then the orbit estimate derived therefrom with two-body dynamics may be poor.

39.1.2 Measurement Outliers

IOD methods present a measurement editing problem. Gross editing is performed on raw tracking data, but exposure to data outliers remains. Effective and refined measurement editing depends on the use of an existing orbit estimate, and that does not exist. Thus measurement residuals cannot be formed,

and measurement residual editing is not possible. The success of IOD can be judged by the success of a subsequent least squares attempt for differential correction to the IOD estimate. Fortunately, with multiple data time points and multiple IOD data sets, the user is able to attempt an IOD solution many times with independent data sets. IOD editing requires manual interaction between program response and the user.

39.2 Common Equations for IOD

39.2.1 Equation of Motion

The two-body vector equation of motion is given by:

$$\ddot{\mathbf{r}} = -\frac{\mu}{r^3}\mathbf{r} \quad (39.1)$$

39.2.2 Triangle Geometry

Let us denote the Earth-centered tracking station vector with \mathbf{s} . Define the instantaneous range vector $\boldsymbol{\rho}$ with the vector triangle:

$$\boldsymbol{\rho} = \mathbf{r} - \mathbf{s} \quad (39.2)$$

define its length ρ with:

$$\rho = \sqrt{\boldsymbol{\rho} \cdot \boldsymbol{\rho}} \quad (39.3)$$

and define the instantaneous unit range vector \mathbf{L} with:

$$\mathbf{L} = \boldsymbol{\rho}/\rho \quad (39.4)$$

Then:

$$\mathbf{r} = \rho\mathbf{L} + \mathbf{s} \quad (39.5)$$

and:

$$\mathbf{r} \cdot \mathbf{r} = r^2 = \rho^2 + 2\rho\mathbf{L} \cdot \mathbf{s} + s^2 \quad (39.6)$$

Chapter 40

Herrick-Gibbs

Given three distinct sets of range, azimuth and elevation measurements, spacecraft position and velocity vectors are calculated according to the method of Gibbs as extended by Herrick [33].

40.1 Position Vectors

Unit range vectors \mathbf{L}_i with their times t_i , for $i \in \{1, 2, 3\}$, are derived directly from the three sets of azimuth and elevation measurements, as in the section herein on station kinematics. Use Eq. 39.5 to form three spacecraft position vectors \mathbf{r}_i from measured range measurements ρ_i , station position vectors \mathbf{s}_i , and the \mathbf{L}_i .

40.1.1 Taylor's Series

Treat time t_2 as fixed so that $\mathbf{r}_2, \dot{\mathbf{r}}_2, \dots$ are vector time constants. Expand \mathbf{r}_1 and \mathbf{r}_3 in Taylor's series about \mathbf{r}_2 :

$$\mathbf{r}_j = \mathbf{r}_2 + \dot{\mathbf{r}}_2 (t_j - t_2) + \frac{1}{2} \ddot{\mathbf{r}}_2 (t_j - t_2)^2 + \frac{1}{3!} \dddot{\mathbf{r}}_2 (t_j - t_2)^3 + \frac{1}{4!} \mathbf{r}^{(4)}_2 (t_j - t_2)^4 + O\left((t_j - t_2)^5\right) \quad (40.1)$$

where $j \in \{1, 3\}$. Ignore $O\left((t_j - t_2)^5\right)$ to write:

$$\mathbf{r}_j = \mathbf{r}_2 + \dot{\mathbf{r}}_2 (t_j - t_2) + \frac{1}{2} \ddot{\mathbf{r}}_2 (t_j - t_2)^2 + \frac{1}{3!} \dddot{\mathbf{r}}_2 (t_j - t_2)^3 + \frac{1}{4!} \mathbf{r}^{(4)}_2 (t_j - t_2)^4 \quad (40.2)$$

where $j \in \{1, 3\}$. Differentiate Eq. 40.2 twice with respect to t_j :

$$\ddot{\mathbf{r}}_j = \ddot{\mathbf{r}}_2 + \dddot{\mathbf{r}}_2 (t_j - t_2) + \frac{1}{2} \mathbf{r}^{(4)}_2 (t_j - t_2)^2 \quad (40.3)$$

Insert Eq. 39.1 twice into Eq. 40.3:

$$-\mu \frac{\mathbf{r}_j}{r_j^3} = -\mu \frac{\mathbf{r}_2}{r_2^3} + \ddot{\mathbf{r}}_2 (t_j - t_2) + \frac{1}{2} \dddot{\mathbf{r}}_2 (t_j - t_2)^2 \quad (40.4)$$

where $j \in \{1, 3\}$. Solve these two vector equations for $\ddot{\mathbf{r}}_2$ and $\dddot{\mathbf{r}}_2$:

$$\begin{bmatrix} \ddot{\mathbf{r}}_2 \\ \dddot{\mathbf{r}}_2 \end{bmatrix} = \frac{\mu}{\Delta} \begin{bmatrix} (t_3 - t_2)^2 / 2 & -(t_1 - t_2)^2 / 2 \\ -(t_3 - t_2) & (t_1 - t_2) \end{bmatrix} \begin{bmatrix} \mathbf{r}_2 / r_2^3 - \mathbf{r}_1 / r_1^3 \\ \mathbf{r}_2 / r_2^3 - \mathbf{r}_3 / r_3^3 \end{bmatrix} \quad (40.5)$$

where:

$$\Delta = (t_1 - t_2)(t_3 - t_2)(t_3 - t_1) / 2 \quad (40.6)$$

Insert Eq. 40.5 into Eq. 40.2 to solve for $\dot{\mathbf{r}}_2$ and $\ddot{\mathbf{r}}_2$ in terms of \mathbf{r}_i for $i \in \{1, 2, 3\}$. Evaluate the two equations of 40.2 (in terms of $\dot{\mathbf{r}}_2$, $\ddot{\mathbf{r}}_2$, $\ddot{\mathbf{r}}_2$, and $\ddot{\mathbf{r}}_2$) as functions of \mathbf{r}_i for $i \in \{1, 2, 3\}$. Add these two equations appropriately to get:

$$\mathbf{r}_2 = c_1 \mathbf{r}_1 + c_3 \mathbf{r}_3 \quad (40.7)$$

$$\dot{\mathbf{r}}_2 = -d_1 \mathbf{r}_1 + d_2 \mathbf{r}_2 + d_3 \mathbf{r}_3 \quad (40.8)$$

where:

$$c_j = A_j (1 + B_j / r_j^3) / (1 - B_2 / r_2^3), \quad j \in \{1, 3\} \quad (40.9)$$

$$A_1 = \frac{t_{23}}{t_{13}}, \quad A_3 = \frac{t_{12}}{t_{13}} \quad (40.10)$$

$$B_1 = \mu (t_{12}^2 + t_{12} t_{23} - t_{23}^2) / 12 \quad (40.11)$$

$$B_2 = \mu (t_{12}^2 + 3t_{12} t_{23} + t_{23}^2) / 12 \quad (40.12)$$

$$B_3 = \mu (-t_{12}^2 + t_{12} t_{23} + t_{23}^2) / 12 \quad (40.13)$$

$$t_{12} = t_2 - t_1, \quad t_{13} = t_3 - t_1, \quad t_{23} = t_3 - t_2 \quad (40.14)$$

and where:

$$d_i = G_i + H_i / r_i^3, \quad i \in \{1, 2, 3\} \quad (40.15)$$

$$G_1 = \frac{t_{23}^2}{t_{12} t_{23} t_{13}}, \quad G_3 = \frac{t_{12}^2}{t_{12} t_{23} t_{13}}, \quad G_2 = G_1 - G_3 \quad (40.16)$$

$$H_1 = \mu t_{23} / 12, \quad H_3 = \mu t_{12} / 12, \quad H_2 = H_1 - H_3 \quad (40.17)$$

Note: we use \mathbf{r}_2 as initially derived from Eq. 39.5, rather than from Eq. 40.7. This proves to be successful for radar range measurements and geocentric orbits. Herrick argues that $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$ must be forced to be coplanar according to Eq. 40.7. Of course $\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3$ are not coplanar due to non two-body perturbative forces.

Chapter 41

Gooding

41.1 Description

41.1.1 Guess Range Values

Given tracking platform positions, GoodingIOD calculates orbit solutions that complete the first and third unit range vectors for which pairs of angles are known but ranges are missing. From each such solution the inertial components of position and velocity are automatically implied.

Components of unambiguous unit range vector estimates \mathbf{L}_1 , \mathbf{L}_2 and \mathbf{L}_3 are calculated directly from measured angles pairs at times t_1 , t_2 , and t_3 . Unit vectors \mathbf{L}_j , $j \in \{1, 2, 3\}$, are accurate three-dimensional direction pointers from tracking platform to spacecraft, but values for the lengths (ranges) ρ_j of the range vectors $\boldsymbol{\rho}_j = \rho_j \mathbf{L}_j$ are unknown. Initial guesses $\rho_1^{(0)}$ and $\rho_3^{(0)}$ are defined by the user, or by program default. Vectors \mathbf{L}_1 and \mathbf{L}_3 are then multiplied by initial range value guesses $\rho_1^{(0)}$ and $\rho_3^{(0)}$ respectively to derive range vector estimates $\boldsymbol{\rho}_1^{(0)}$ and $\boldsymbol{\rho}_3^{(0)}$ at times t_1 and t_3 . Vectors $\boldsymbol{\rho}_1^{(0)}$ and $\boldsymbol{\rho}_3^{(0)}$ are correctly directed, but their length estimates may be poor. Vectors $\boldsymbol{\rho}_1^{(0)}$ and $\boldsymbol{\rho}_3^{(0)}$ are added to known and accurate platform position vectors \mathbf{s}_1 and \mathbf{s}_3 respectively at times t_1 and t_3 to provide spacecraft position vector estimates $\mathbf{r}_1^{(0)} = \mathbf{s}_1 + \boldsymbol{\rho}_1^{(0)}$ and $\mathbf{r}_3^{(0)} = \mathbf{s}_3 + \boldsymbol{\rho}_3^{(0)}$ at times t_1 and t_3 . Errors in the spacecraft position vector estimates $\mathbf{r}_1^{(0)}$ and $\mathbf{r}_3^{(0)}$ may be large, in both direction and size. Even so, we have two spacecraft position vector estimates $\mathbf{r}_1^{(0)}$ and $\mathbf{r}_3^{(0)}$ at two distinct times t_1 and t_3 .

41.1.2 Lambert

The unknown central angle θ between true position vectors \mathbf{r}_1 and \mathbf{r}_3 , in the plane of \mathbf{r}_1 and \mathbf{r}_3 , is related to the orbit period P , and measurement time-tag difference $(t_3 - t_1)$. For each value of θ , and for integers $k \in \{0, 1, 2, 3, \dots\}$, define k such that $(k)\pi < \theta \leq (k+1)\pi$.

Examples: When $0 < \theta \leq \pi$, then $k = 0$, and the central angle θ is completely contained within the first one-half orbit. When $\pi < \theta \leq 2\pi$, then $k = 1$, and the central angle θ is completely contained within one orbit. Thus k is the number of half-orbits for which the central angle θ , between position vectors \mathbf{r}_1 and \mathbf{r}_3 , is completely contained. When the orbit is circular, the relation between $(t_3 - t_1)$, orbit period P , and k is simple, and the user can make a good guess for k (see Table 41.2). But when the orbit is highly eccentric, this relation is not obvious.

Lambert Problem

Given values for components of any two distinct spacecraft position vectors \mathbf{r}_1 and \mathbf{r}_3 , and their times t_1 and t_3 , calculate the spacecraft velocity $\dot{\mathbf{r}}_1$ at time t_1 , or calculate the spacecraft velocity $\dot{\mathbf{r}}_3$ at time t_3 .

Multiple Solutions

When $k = 0$ or $k = 1$, Lambert's Problem has exactly one solution, but when $k \geq 2$, Lambert's Problem, like a quadratic equation, has two solutions, no solution, or one solution in a limiting case that occurs with probability zero. Indicator $i = 0$ if $k \in \{0, 1\}$, but $i \in \{0, 1\}$ if $k \geq 2$. This Indicates which of two Lambert solutions are to be used, if there are two.

Universal Variables

Gooding's implementation of the Lambert Problem solution is an accurate, two-iteration universal one, valid for all conic orbits: ellipse, parabola, and hyperbola.

41.1.3 Gooding

The GoodingIOD algorithm submits $\mathbf{r}_1^{(0)}$ and $\mathbf{r}_3^{(0)}$, with times t_1 and t_3 , to the Lambert algorithm to calculate velocity $\dot{\mathbf{r}}_1^{(0)}$. Although the velocity vector $\dot{\mathbf{r}}_1^{(0)}$ is likely a poor estimate of $\dot{\mathbf{r}}_1$, at least we have a complete orbit estimate from $\mathbf{r}_1^{(0)}$ and $\dot{\mathbf{r}}_1^{(0)}$. This enables the propagation of $\mathbf{r}_1^{(0)}$ and $\dot{\mathbf{r}}_1^{(0)}$ to time t_2 to obtain estimates $\mathbf{r}_2^{(0)}$ (and $\dot{\mathbf{r}}_2^{(0)}$). Calculate $\boldsymbol{\rho}_2^{(0)} = \mathbf{r}_2^{(0)} - \mathbf{s}_2$, an erroneous estimate of the true range vector $\boldsymbol{\rho}_2$. Recall that \mathbf{L}_2 (from angles data) is a good estimate of the unit range vector at time t_2 , so project $\boldsymbol{\rho}_2^{(0)}$ onto a plane orthogonal to \mathbf{L}_2 to define two components of the error in $\boldsymbol{\rho}_2^{(0)}$. Now correct $\rho_1^{(0)}$ and $\rho_3^{(0)}$ to reduce the two components of error in $\boldsymbol{\rho}_2^{(0)}$, and generate improved estimates $\rho_1^{(1)}$ and $\rho_3^{(1)}$ of range, thereby also calculating improved estimates $\mathbf{r}_1^{(1)}$ and $\dot{\mathbf{r}}_1^{(1)}$ of position and velocity. This procedure is iterated on integer i so as to drive the two components of error in $\boldsymbol{\rho}_2^{(i)}$ to zero, thereby deriving best estimates of ρ_1 , ρ_3 , \mathbf{r}_2 , and $\dot{\mathbf{r}}_2$.

When the embedded Lambert algorithm provides a solution for the spacecraft velocity $\dot{\mathbf{r}}_1$ at time t_1 , then GoodingIOD converges to a *local* solution for

\mathbf{r}_2 and $\dot{\mathbf{r}}_2$. But the price for extraordinary initial errors in $\rho_1^{(0)}$ and $\rho_3^{(0)}$ is that the *local* solution for \mathbf{r}_2 and $\dot{\mathbf{r}}_2$ may not be useful.

Multiple Solutions

When $k = 0$, there do formally exist either 0, 1, 2, or 3 distinct solutions to the GoodingIOD problem. Unfortunately, there is no sure technique to identify the number of distinct solutions a priori. When there are 2 or 3 distinct solutions, only one of them is useful, but we have found that in the large majority of cases tested, that one is found. When there are formally 3 distinct solutions, two of them may be complex, and will not be found, because the algorithm is restricted to the real domain.

As k is increased, the number of solutions also increases, though only up to some maximum value. If the value of k is not known in advance, therefore, the search for the useful solution can get increasingly complicated.

41.2 A Priori Orbit Information

When there are multiple distinct solutions, only one of them will be useful – associated with the true orbit. It is necessary to correctly couple the selections of the three measurement pair time-tags t_1 , t_2 , and t_3 with the *HalfRevEstimate* = k from the set $\{0, 1, 2, 3, \dots\}$ in order to find the useful solution. One way to do this is to search through $k \in \{0, 1, 2, 3, \dots\}$ beginning with $k = 0$. But when the user has useful a priori information (*e.g.*, orbit class LEO, MEO, GEO, HEO, orbit period P , semi-major axis a , or eccentricity e), then the search for the useful solution may be simplified. Table 41.1 correlates orbit class roughly with P (min), P (sec), a (km), and e .

41.2.1 Near-Circular Orbits

With eccentricity zero (or near zero), any uniform partition of orbit period P maps uniformly (or almost uniformly) to increments in central angle – true argument of latitude differences. This enables association of orbit period P with the measurement time-tag difference $t_3 - t_1$ and the *HalfRevEstimate* = k . For single-station IOD¹, the simplest approach is to choose $k = 0$, then select measurements such that $0 < t_3 - t_1 < P/2$, but such that as much of the interval $[t_1, t_3]$ is used as possible in this selection so as to attenuate the effects of measurement white noise. Table 41.2 correlates k with P and $t_3 - t_1$.

¹Double-station IOD is preferred for accuracy.

41.2.2 High Eccentricity Orbits

Given an a priori estimate for orbit period P , and knowing that the eccentricity e is large (e.g., $e > 0.7$), one may choose time-tags t_1 , t_2 , and t_3 such that $0 < t_3 - t_1 < P$. Then one must perform two sets of tests with *HalfRevEstimate* = k , for $k = 0$, and for $k = 1$.

41.3 White Noise

Gauss[26] discovered white noise² on telescopic angles measurements in right ascension and declination in 1795, while performing orbit determination. All real angles measurements have white noise³ that degrades all algorithms⁴ for angles-only IOD. For single-station single-pass LEO tracking GoodingIOD overcomes the white-noise problem when $t_3 - t_1 > 0$ is sufficiently large. For two-station LEO tracking, GoodingIOD significantly reduces position and velocity errors due to white-noise, relative to single-station tracking, if all measurements are selected from a common six minute time interval and $t_3 - t_1 = 1$ min approximately.

41.4 Tropospheric Effects

Degrading effects of the troposphere, from ground based platforms, can be significantly reduced by selecting measurements whose elevations are greater than 10 deg.

41.5 Inspection of Kepler Orbit Element Values

Given a priori information regarding the orbit, and given multiple solutions from the same measurement set, the user can frequently eliminate useless solutions by inspection of the Kepler orbit elements, particularly the semi-major axis and eccentricity.

41.6 Multiple Solutions from Each Set of Distinct Measurement Sets

Comparison of sets of multiple solutions from distinct measurement sets may, or may not, identify the useful solution.

²Gauss[26] invoked the normal density function to model white noise, and invented the iterative least squares algorithm (1795) to perform orbit determination on an overdetermined set of telescopic measurements in right ascension and declination.

³Practically, a realizable white noise time sequence may be visualized as a random normal sequence where every component is independent of all others (no serial correlation), and where the ensemble variance is finite.

⁴The classical LaPlace IOD method is destroyed by white noise if its RMS is significant, when applied to geocentric orbits .

41.7 Least Squares

Given multiple solutions from the same measurement set, the user can test each solution by running least squares on it with an overdetermined set (more than three) of angles measurements. Least squares will converge only on the useful solution.

Orbit Class	P (min)	P (sec)	a (km)	e
LEO	100	6000	7137	$0.000 < e < 0.050$
GEO	1436	86169	42163	$0.000 < e < 0.001$
GPS (MEO)	718	43080	26560	$0.000 < e < 0.001$
HEO	718	43080	26561	0.7 (Molniya)

Table 41.1: Mean Orbit Period vs Orbit Class

$t_3 - t_1$ vs P	k
$0 < t_3 - t_1 < P/2$	0
$P/2 < t_3 - t_1 < P$	1
$P < t_3 - t_1 < (3/2)P$	2
$(3/2)P < t_3 - t_1 < 2P$	3
\vdots	\vdots

Table 41.2: Orbit Period Relations

Part X

Tracking and Data Relay Satellite System (TDRSS)

Chapter 42

TDRSS Range and Doppler

Simultaneous orbit determination of multiple TDRS (GEO) orbits and multiple LEOs and/or MEOs is enabled with the implementation of capabilities to process four-legged range and five legged Doppler measurements for NASA's Telemetry and Data Relay Satellite System (TDRSS). NASA's associated Bilateral Ranging Transponder System (BRTS) range and Doppler measurements can also be processed to perform orbit determination on each TDRS separately, or on all TDRS' simultaneously by fusing BRTS measurements with TDRSS measurements. Radio signals for measurements in the same round-trip link are all generated by the same atomic ground clock, and are said to be *coherent* because the effects of clock phenomenology and relativity are subtracted out. That is, the four-legged range and five legged Doppler measurements are two-way measurements.

Both of the two time-varying transponder biases in the four legged range link may be estimated, separately¹ or simultaneously, with ground station range biases. Also, time-varying range-sum biases can be estimated to include the summed effects of transponder biases and ground station biases.

42.1 TDRSS Range Vectors

TDRS refers to a Tracking and Data Relay Satellite. The Tracking and Data Relay Satellite System (TDRSS) consists of a constellation of geosynchronous TDRS, a constellation of LEO spacecraft, the White Sands Ground Terminal (WSGT), associated hardware and software, and personnel. Fig. 42.1 depicts the Earth centered positions of the WSGT at times t_0 and t_4 with position vectors \mathbf{S}_0 and \mathbf{S}_4 . Consider a radio plane wave that is emitted from the WSGT at time t_0 , received and retransmitted by a TDRS transponder at time t_1 , received and retransmitted by a LEO spacecraft transponder at time t_2 , received and retransmitted by the TDRS transponder at time t_3 , and received by the

¹Time-constant values for the two transponder biases in the four legged range link were shown to be simultaneously observable when ground station range biases are known a priori.

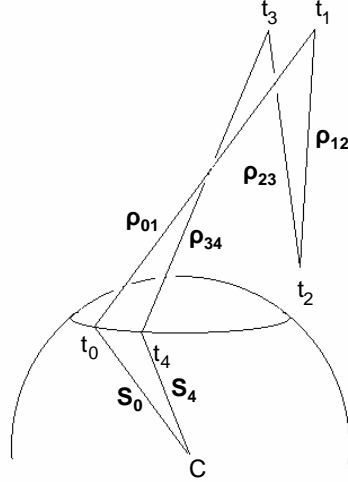


Figure 42.1: Four TDRSS Range Vectors

WSGT at time t_4 . Fig. 42.1 identifies associated range vectors ρ_{01} , ρ_{12} , ρ_{23} , and ρ_{34} . Let us denote TDRS position vectors by \mathbf{R}_j at t_j , and LEO spacecraft position vectors by \mathbf{r}_j at t_j , for $j \in \{1, 2, 3\}$. Define spacecraft position vectors:

$$\mathbf{R}_1 = \mathbf{S}_0 + \rho_{01} \quad (42.1)$$

$$\mathbf{r}_2 = \mathbf{R}_1 + \rho_{12} \quad (42.2)$$

$$\mathbf{R}_3 = \mathbf{r}_2 + \rho_{23} \quad (42.3)$$

and the WSGT vector constraint:

$$\mathbf{S}_4 = \mathbf{R}_3 + \rho_{34} \quad (42.4)$$

42.2 Range Sum

The following construction assumes that all range effects associated with cable delay, electromagnetic phase center, Faraday rotation, relative rotation between antennas, transponders, troposphere, and ionosphere are addressed elsewhere.

42.2.1 Range Elements

Define:

$$t_{(j)(j+1)} = t_{j+1} - t_j, \quad t_j < t_{j+1}, \quad j \in \{0, 1, 2, 3\} \quad (42.5)$$

Define range elements:

$$\rho_{01} = \sqrt{\rho_{01} \cdot \rho_{01}} = ct_{01} \quad (42.6)$$

$$\rho_{12} = \sqrt{\rho_{12} \cdot \rho_{12}} = ct_{12} \quad (42.7)$$

$$\rho_{23} = \sqrt{\rho_{23} \cdot \rho_{23}} = ct_{23} \quad (42.8)$$

$$\rho_{34} = \sqrt{\rho_{34} \cdot \rho_{34}} = ct_{34} \quad (42.9)$$

Each range element is defined by two expressions, a geometric expression and a radio wave propagation time delay expression.

42.2.2 Range Sum Definition

Let us denote the two-way range sum with ρ_{0L} , and define it with:

$$\rho_{0L} = \rho_{01} + \rho_{12} + \rho_{23} + \rho_{34} = ct_{0L} \quad (42.10)$$

where:

$$L = 4 \quad (42.11)$$

42.2.3 Range Sum Time Tag

The range sum time tag is defined by $t_L = t_4$.

42.2.4 Construction of the Range Sum Representation

Range Iterations

Given the time-tag $t_L = t_4$, we wish to calculate a first approximation $\tilde{\rho}_{34}$ for ρ_{34} . Evaluate \mathbf{S}_4 at $t_L = t_4$. Let us denote a first approximation for time difference t_{34} with \tilde{t}_{34} and set $\tilde{t}_{34} = 0.133s$. Then from Eq. 42.5:

$$\tilde{t}_3 = t_4 - \tilde{t}_{34}$$

Evaluate $\tilde{\mathbf{R}}_3$ at \tilde{t}_3 . Rearrange Eq. 42.4 to approximate the range vector ρ_{34} :

$$\tilde{\rho}_{34} = \mathbf{S}_4 - \tilde{\mathbf{R}}_3$$

Use Eq. 42.9 to calculate the range approximation:

$$\tilde{\rho}_{34} = \sqrt{\tilde{\rho}_{34} \cdot \tilde{\rho}_{34}}$$

and use Eq. 42.9 again to calculate a second approximation for $t_{3,4}$:

$$\bar{t}_{3,4} = \tilde{\rho}_{34}/c$$

Set $\tilde{t}_{3,4} = \bar{t}_{3,4}$ and iterate on the above sequence until:

$$|\tilde{t}_{3,4} - \bar{t}_{3,4}| < \epsilon_{3,4}$$

to yield t_3 , t_{34} , and ρ_{34} . Three iterations are usually sufficient.

Given the time t_3 , calculate t_2 , t_{23} , and ρ_{23} as above. Given the time t_2 , calculate t_1 , t_{12} , and ρ_{12} as above. Given the time t_1 , calculate t_0 , t_{01} , and ρ_{01} as above.

Range Sum

Now use Eq. 42.10 to calculate the range sum ρ_{0L} .

42.3 Range Sum Partial Derivatives

42.3.1 Eliminate Light-Time

Simple and sufficient measurement to state partial derivative approximations are achieved by eliminating light-time effects:

$$\rho_1 \equiv \rho_{01} = -\rho_{34} \quad (42.12)$$

$$\rho_2 \equiv \rho_{12} = -\rho_{23} \quad (42.13)$$

$$\mathbf{S} \equiv \mathbf{S}_0 = \mathbf{S}_4 \quad (42.14)$$

$$\mathbf{R} \equiv \mathbf{R}_1 = \mathbf{R}_3 \quad (42.15)$$

$$\mathbf{r} \equiv \mathbf{r}_2 \quad (42.16)$$

$$\rho_1 = \sqrt{\rho_1 \cdot \rho_1} \quad (42.17)$$

$$\rho_2 = \sqrt{\rho_2 \cdot \rho_2} \quad (42.18)$$

all defined at the same epoch $t = t_4$.

42.3.2 Simplified Expressions

Then:

$$\mathbf{r} = \mathbf{S} + \boldsymbol{\rho}_1 + \boldsymbol{\rho}_2 = \mathbf{R} + \boldsymbol{\rho}_2 \quad (42.19)$$

$$\mathbf{R} = \mathbf{S} + \boldsymbol{\rho}_1 \quad (42.20)$$

Define the one-way range sum approximation with:

$$\rho = \rho_1 + \rho_2 = \sqrt{\boldsymbol{\rho}_1 \cdot \boldsymbol{\rho}_1} + \sqrt{\boldsymbol{\rho}_2 \cdot \boldsymbol{\rho}_2} \quad (42.21)$$

42.3.3 Differentials

Differentiate Eqs. 42.19 and 42.20:

$$\delta \mathbf{r} = \delta \mathbf{S} + \delta \boldsymbol{\rho}_1 + \delta \boldsymbol{\rho}_2 = \delta \mathbf{R} + \delta \boldsymbol{\rho}_2 \quad (42.22)$$

$$\delta \mathbf{R} = \delta \mathbf{S} + \delta \boldsymbol{\rho}_1 \quad (42.23)$$

Differentiate Eq. 42.21:

$$\delta \rho = \hat{\boldsymbol{\rho}}_1 \cdot \delta \boldsymbol{\rho}_1 + \hat{\boldsymbol{\rho}}_2 \cdot \delta \boldsymbol{\rho}_2 \quad (42.24)$$

where:

$$\hat{\boldsymbol{\rho}}_1 = \boldsymbol{\rho}_1 / \rho_1 \quad (42.25)$$

$$\hat{\boldsymbol{\rho}}_2 = \boldsymbol{\rho}_2 / \rho_2 \quad (42.26)$$

42.3.4 LEO Satellite Range Sum Partial

Vary \mathbf{r} , but with \mathbf{S} and \mathbf{R} held fixed. Then:

$$\delta \mathbf{S} = \delta \mathbf{R} = \mathbf{0} \quad (42.27)$$

And from Eq. 42.23:

$$\delta \boldsymbol{\rho}_1 = \mathbf{0} \quad (42.28)$$

According to Eq. 42.22:

$$\delta \mathbf{r} = \delta \boldsymbol{\rho}_2 \quad (42.29)$$

and from Eq. 42.24:

$$\delta \rho = \hat{\boldsymbol{\rho}}_2 \cdot \delta \mathbf{r} \quad (42.30)$$

Now $\delta \mathbf{r}$ has the decomposition on the inertial frame:

$$\delta \mathbf{r} = \sum_{j=1}^3 (\delta \mathbf{r} \cdot \mathbf{i}_j) \mathbf{i}_j = \sum_{j=1}^3 \delta z_j \mathbf{i}_j \quad (42.31)$$

where:

$$\delta z_j = \delta \mathbf{r} \cdot \mathbf{i}_j \quad (42.32)$$

Note that:

$$\delta \rho = \sum_{j=1}^3 \frac{\partial \rho}{\partial z_j} \delta z_j \quad (42.33)$$

So insert Eq. 42.31 into Eq. 42.30:

$$\delta \rho = \sum_{j=1}^3 (\hat{\boldsymbol{\rho}}_2 \cdot \mathbf{i}_j) \delta z_j = \sum_{j=1}^3 \frac{\partial \rho}{\partial z_j} \delta z_j \quad (42.34)$$

from which:

$$\frac{\partial \rho}{\partial z_j} = \hat{\boldsymbol{\rho}}_2 \cdot \mathbf{i}_j, \quad j \in \{1, 2, 3\} \quad (42.35)$$

The LEO position partials are just the inertial components of the unit range vector $\hat{\boldsymbol{\rho}}_2$, where from Eq. 42.19:

$$\boldsymbol{\rho}_2 = \mathbf{r} - \mathbf{R} \quad (42.36)$$

$$\hat{\boldsymbol{\rho}}_2 = \boldsymbol{\rho}_2 / \rho_2 \quad (42.37)$$

Define the 1×3 matrix $\partial \rho / \partial z$:

$$\frac{\partial \rho}{\partial z} = \left[\frac{\partial \rho}{\partial z_1} \quad \frac{\partial \rho}{\partial z_2} \quad \frac{\partial \rho}{\partial z_3} \right] \quad (42.38)$$

Instantaneously, variations in $\dot{\mathbf{r}}$ leave \mathbf{r} invariant, thus:

$$\frac{\partial \rho}{\partial \dot{z}} = 0_{1 \times 3} \quad (42.39)$$

For LEO, define:

$$Z = \begin{bmatrix} z \\ \dot{z} \end{bmatrix} \quad (42.40)$$

Then the 1×6 matrix of range sum with respect to LEO ECI position and velocity partials is defined by:

$$\frac{\partial \rho}{\partial Z} = \left[\frac{\partial \rho}{\partial z} \quad \frac{\partial \rho}{\partial \dot{z}} \right] \quad (42.41)$$

42.3.5 TDRS Range Sum Partial

Vary \mathbf{R} , but with \mathbf{S} and \mathbf{r} held fixed. Then:

$$\delta\mathbf{S} = \delta\mathbf{r} = \mathbf{0} \quad (42.42)$$

and from Eq. 42.23:

$$\delta\rho_1 = \delta\mathbf{R} \quad (42.43)$$

From Eq. 42.22:

$$\delta\rho_2 = -\delta\rho_1 \quad (42.44)$$

So:

$$\delta\rho_2 = -\delta\mathbf{R} \quad (42.45)$$

Then from Eq. 42.24:

$$\delta\rho = (\hat{\rho}_1 - \hat{\rho}_2) \cdot \delta\mathbf{R} \quad (42.46)$$

Now $\delta\mathbf{R}$ has the decomposition on the inertial frame:

$$\delta\mathbf{R} = \sum_{j=1}^3 (\delta\mathbf{R} \cdot \mathbf{i}_j) \mathbf{i}_j = \sum_{j=1}^3 \delta z'_j \mathbf{i}_j \quad (42.47)$$

where:

$$\delta z'_j = \delta\mathbf{R} \cdot \mathbf{i}_j \quad (42.48)$$

Note that:

$$\delta\rho = \sum_{j=1}^3 \frac{\partial\rho}{\partial z'_j} \delta z'_j \quad (42.49)$$

So insert Eq. 42.47 into Eq. 42.46:

$$\delta\rho = \sum_{j=1}^3 [(\hat{\rho}_1 - \hat{\rho}_2) \cdot \mathbf{i}_j] \delta z'_j = \sum_{j=1}^3 \frac{\partial\rho}{\partial z'_j} \delta z'_j \quad (42.50)$$

from which:

$$\frac{\partial\rho}{\partial z'_j} = (\hat{\rho}_1 - \hat{\rho}_2) \cdot \mathbf{i}_j, \quad j \in \{1, 2, 3\} \quad (42.51)$$

The TDRS position partials are just the inertial components of the range vector difference $(\hat{\rho}_1 - \hat{\rho}_2)$. Define the 1×3 matrix $\partial\rho/\partial z$:

$$\frac{\partial\rho}{\partial z'} = \left[\begin{array}{ccc} \frac{\partial\rho}{\partial z'_1} & \frac{\partial\rho}{\partial z'_2} & \frac{\partial\rho}{\partial z'_3} \end{array} \right] \quad (42.52)$$

Instantaneously, variations in $\dot{\mathbf{R}}$ leave \mathbf{R} invariant, thus:

$$\frac{\partial \rho}{\partial \dot{z}'} = \mathbf{0}_{1 \times 3} \quad (42.53)$$

For TDRS, define:

$$\mathbf{Z}' = \begin{bmatrix} \dot{z}' \\ z' \end{bmatrix} \quad (42.54)$$

Then the 1×6 matrix of range sum with respect to TDRS ECI position and velocity partials is defined by:

$$\frac{\partial \rho}{\partial \mathbf{Z}'} = \begin{bmatrix} \frac{\partial \rho}{\partial z'} & \frac{\partial \rho}{\partial \dot{z}'} \end{bmatrix} \quad (42.55)$$

42.4 Doppler Measurements

Define symbols:

f_T	reference transmit frequency
f_D	instantaneous Doppler frequency
f_P	pilot tone frequency
c	speed of light in vacua
$\dot{\rho}_{01}$	leg 1 instantaneous range-rate
$\dot{\rho}_{12}$	leg 2 instantaneous range-rate
$\dot{\rho}_{23}$	leg 3 instantaneous range-rate
$\dot{\rho}_{34}$	leg 4 instantaneous range-rate
$\dot{\rho}_P$	instantaneous pilot tone range-rate

Table 42.1: Doppler Measurement Symbol Definitions

The Doppler frequency f_D extracted from the TDRSS receiver hardware has the form:

$$f_D = -\frac{1}{c} \left[f_T \sum_{j=1}^4 \dot{\rho}_{(j-1)(j)} + f_P \dot{\rho}_P \right] \quad (42.56)$$

Integrate Eq. 42.56 across the carrier phase integration interval $[t_L, t_{L+1}]$, then divide through by $(t_{L+1} - t_L)$, to define the TDRSS Doppler measurement representation \bar{f}_D :

$$\bar{f}_D \equiv -\frac{1}{t_{L+1} - t_L} \int_{t_L}^{t_{L+1}} f_D dt$$

$$= -\frac{1}{c(t_{L+1} - t_L)} \left[f_T \sum_{j=1}^4 \int_{t_L}^{t_{L+1}} \dot{\rho}_{(j-1)(j)} dt + f_P \int_{t_L}^{t_{L+1}} \dot{\rho}_P dt \right] \quad (42.57)$$

Each integral on the right hand side has the form:

$$\int_{t_L}^{t_{L+1}} \dot{\rho} dt = \int_{t_L}^{t_{L+1}} d\rho = \rho(t_{L+1}) - \rho(t_L) \quad (42.58)$$

With these integrations, the TDRSS Doppler measurement representation \bar{f}_D (Eq. 42.57) becomes:

$$\bar{f}_D = -\frac{1}{c(t_{L+1} - t_L)} \left[f_T \sum_{j=1}^4 \Delta\rho_{(j-1)(j)} + f_P \Delta\rho_P \right] \quad (42.59)$$

where:

$$\Delta\rho_{(j-1)(j)} = \rho_{(j-1)(j)}(t_{L+1}) - \rho_{(j-1)(j)}(t_L) \quad (42.60)$$

$$\Delta\rho_P = \rho_p(t_{L+1}) - \rho_p(t_L) \quad (42.61)$$

42.5 DopplerPartials

The time tags for Doppler measurements are at t_{L+1} , and the orbit estimate:

$$Z(t_{L+1}) = \begin{bmatrix} z(t_{L+1}) \\ \dot{z}(t_{L+1}) \end{bmatrix}$$

will sit nominally at t_{L+1} . Thus the first term in the first and second terms of Eqs. 42.63 and 42.64 is already calculated for range sum partials. The second term in the first and second terms of Eqs. 42.63 and 42.64 will require the use of the linear transition function $\Phi(t_L, t_{L+1})$ to move the calculation of the 1×6 matrix $\partial\rho(t_L)/\partial Z(t_L)$ from time t_L to time t_{L+1} :

$$\frac{\partial\rho(t_L)}{\partial Z(t_{L+1})} = \frac{\partial\rho(t_L)}{\partial Z(t_L)} \frac{\partial Z(t_L)}{\partial Z(t_{L+1})} = \frac{\partial\rho(t_L)}{\partial Z(t_L)} \Phi(t_L, t_{L+1}) \quad (42.62)$$

42.5.1 LEO Satellite Doppler Partial

Differentiate Eq. 42.59:

$$\frac{\partial\bar{f}_D}{\partial z} = -\frac{1}{c(t_{L+1} - t_L)} \left[f_T \sum_{j=1}^4 \frac{\partial\rho_{(j-1)(j)}(t_{L+1}) - \rho_{(j-1)(j)}(t_L)}{\partial z(t_{L+1})} + f_P \frac{\partial\rho_p(t_{L+1}) - \rho_p(t_L)}{\partial z(t_{L+1})} \right] \quad (42.63)$$

42.5.2 TDRS Doppler Partial

Differentiate Eq. 42.59:

$$\frac{\partial \bar{f}_D}{\partial z'} =$$

$$-\frac{1}{c(t_{L+1} - t_L)} \left[f_T \sum_{j=1}^4 \frac{\partial \rho_{(j-1)(j)}(t_{L+1}) - \rho_{(j-1)(j)}(t_L)}{\partial z'(t_{L+1})} + f_P \frac{\partial \rho_p(t_{L+1}) - \rho_p(t_L)}{\partial z'(t_{L+1})} \right]$$

(42.64)

Chapter 43

One-Way Return-Link Doppler

The TDRSS one-way return-link three-legged Doppler measurement filter capability has been implemented according to GSFC documentation.

Part XI

Global Positioning System (GPS) Low Earth Orbit (LEO) Receivers

Chapter 44

Pseudo-Range Filtering

44.1 Introduction

The USER spacecraft pseudo-range measurement representation chain (Eq. 44.1) is adopted from JPL[132], but the components in that chain are inappropriate, in part, due to JPL modeling of the USER clock as a ground fixed clock. Also, the JPL document does not derive relativistic expressions from first principles. This makes it difficult to validate the JPL equations. We have derived appropriate expressions from the Schwarzschild metric that enable the reader to validate the equations we have provided.

Measurement-to-state analytical partial derivative approximations for USER GPS psuedo-range measurements are derived and given by Eqs. 44.81 through 44.86.

The USER stochastic clock model is derived from Allan[1]. From this clock model, the filter GPS USER clock propagation, transition, covariance propagation, and process noise covariance functions are derived and given in Eqs. 34.35, 34.36, 34.37, and 34.38 respectively.

The synchronization of USER spacecraft clock to NAVSTAR SV clock is discussed.

44.1.1 Pseudo-Range Option

“A properly equipped receiver will detect amplitude, pseudo-range and phase measurements for each of the C/A, L1 P-code (P1) and L2 P-code (P2) signals. The C/A and P1 measurements essentially contain identical information, however C/A is preferred over P1 because its power is stronger by 3 dB and is not encrypted. Therefore, the basic observables used during an occultation experiment are the C/A phase and the P2 phase measurements between the low-Earth orbit (LEO) satellite and the occulting GPS satellite” (Kursinski [57]). This conclusion applies as well to orbit determination. Therefore, we should provide an option to remove ionosphere with either {P1,P2} or {C/A,P2}.

44.2 USER GPS Pseudo-Range

44.2.1 Pseudo-Range

Let us denote the pseudo-range measurement representation by ρ , and define it with:

$$\begin{aligned}\rho &= c [(T^3 - \tau^3) + (\tau^3 - t^3) + (t^3 - t^2) + (t^2 - \tau^2) + (\tau^2 - T^2)] + \beta \\ &= c [T^3 - T^2] + \beta\end{aligned}\quad (44.1)$$

where c is speed of light in vacuo, β is a stored constant (range bias), and time notations are identified below.

44.2.2 Range Measurement Time Notation

T^3	receiver time tag
τ^3	receiver proper time
t^3	receiver coordinate time (TDT)
t^2	transmitter coordinate time (TDT)
τ^2	transmitter proper time
T^2	transmitter time tag

44.2.3 Time Differences

$T^3 - \tau^3$	receiver clock error (user spacecraft clock)
$\tau^3 - t^3$	receiver proper time - receiver coordinate time
$t^3 - t^2$	receiver coordinate time - transmitter coordinate time
$t^2 - \tau^2$	transmitter coordinate time - transmitter proper time
$\tau^2 - T^2$	transmitter clock error (GPS NAVSTAR clock)

44.2.4 Receiver Clock Errors

The receiver clock error $T^3 - \tau^3$ will be composed of two additive components: a stored constant bias scalar β_x and a clock phase state estimate parameter x , defined by Eqs. 34.23 and 34.13. The generalized derivative $y = dx/dt$, defined by Eqs. 34.29 and 34.13, will also be estimated, and is associated with an additive constant bias scalar β_y . The coupled errors in the two-parameter clock state estimate will be accounted for with a 2x2 state estimate clock error covariance. The 2x2 clock transition matrix function is defined by Eq. 34.36, and the 2x2 clock process error covariance matrix function is defined by Eq. 34.37.

Any long term estimated constant biases (*e.g.*, clock synchronization error and drift) in x and y will be transferred manually to the unestimated parameters β_x and β_y .

The state estimate is composed of clock modeling effects, and non-clock modeling effects – non-separable from clock modeling effects.

Clock modeling effects

- Synchronization error between receiver clock and transmitter clock (assumed here to be due to the receiver clock)
- Stochastic phenomenology of the receiver clock (assumed here to be inferior to the NAVSTAR clock)
- NAVSTAR transmitter clock error due to modeling and propagation errors in the coefficients downlinked in the navigation message

Non-clock modeling effects due to modeling errors

- NAVSTAR propagation errors in position
- Antenna multipath (spacecraft body specific)
- Antenna phase center location
- Mutual rotations of transmitting and receiving antennas
- Troposphere
- Ionosphere
- Faraday rotation
- Unmodeled cable lengths in the hardware

44.2.5 Receiver Proper Time less Receiver Coordinate Time (User Spacecraft)

For this subsection we shall drop the superscript 3 that identifies the receiver and add a subscript j that distinguishes time. Thus: $\tau^3 = \tau_j$ and $t^3 = t_j$, for $j \in \{k, k+1\}$. From 44.28, define the sequential filter time-update calculation:

$$(\tau_{k+1} - t_{k+1}) = (\tau_k - t_k) - (t_{k+1} - t_k) \frac{3\mu}{2ac^2} - \frac{2}{c^2} (r_{k+1}\dot{r}_{k+1} - r_k\dot{r}_k) \quad (44.2)$$

where $k \in \{0, 1, 2, \dots\}$. The relation between τ_0 and t_0 is arbitrary. For initialization¹ we choose to define:

$$\tau_0 = t_0 \quad (44.3)$$

The relativistic constant clock phase bias, in the User spacecraft clock, is contained in its proper time τ_0 at initialization of sequential orbit determination.

¹The JPL initialization convention is defined by:

$$\tau_0 - t_0 = -[(TAI - UTC) + (TDT - TAI)]$$

This is of course relevant only for receivers that do not perform clock steering. For in this case, GPS range measurements from different receiver channels must be differenced at the same instant to remove the effects of clock steering, and thus remove all User clock effects for orbit determination.

44.2.6 Coordinate Time Difference

Coordinate times are in TDT. The coordinate time difference ($t^3 - t^2$) is associated with the positional coordinate differences of ECI ICRF ephemerides, and is given:

$$t^3 - t^2 = \frac{r_{32}}{c} + \frac{2\mu}{c^3} \ln \left(\frac{r_3 + r_2 + r_{32}}{r_3 + r_2 - r_{32}} \right) \quad (44.4)$$

This light-time equation is due to Holdridge[37]. The unknown quantity t^2 (coordinate time of transmission) can be calculated by Newton-Raphson ([132] Eq. 4.2.3.1.19 – note mislabel here):

$$t^{2(i+1)} = t^{2(i)} + \left[t^3 - t^{2(i)} - \left(\frac{2\mu}{c^3} \ln \left(\frac{r_3 + r_2^{(i)} + r_{32}^{(i)}}{r_3 + r_2^{(i)} - r_{32}^{(i)}} \right) \right) \right] / \left[1 - \mathbf{r}_{32}^{(i)} \cdot \dot{\mathbf{r}}_2^{(i)} / (cr_{32}^{(i)}) \right] \quad (44.5)$$

where for $i = 0$, and $t^{2(i)} = t^{2(0)}$:

$$t^{2(0)} = t^3$$

44.2.7 Transmitter Coordinate Time less Transmitter Proper Time (NAVSTAR)

For this subsection we shall drop the superscript 2 that identifies the transmitter and add a subscript j that distinguishes time. Thus: $\tau^2 = \tau_j$ and $t^2 = t_j$, for $j \in \{k, k+1\}$. The following equations (derived below) are equivalent:

$$(t_{k+1} - \tau_{k+1}) = (t_k - \tau_k) + (\tau_{k+1} - \tau_k) \frac{3\mu}{2ac^2} + \frac{2}{c^2} (r_{k+1}\dot{r}_{k+1} - r_k\dot{r}_k) \quad (44.6)$$

$$(t_{k+1} - \tau_{k+1}) = (t_k - \tau_k) + (\tau_{k+1} - \tau_k) \frac{3\mu}{2ac^2} + \frac{2e\sqrt{\mu a}}{c^2} (\sin E_{k+1} - \sin E_k) \quad (44.7)$$

The constant clock rate $3\mu/(2ac^2)$ is accounted for by offsetting the frequency of the onboard NAVSTAR oscillator. So set:

$$(\tau_{k+1} - \tau_k) \frac{3\mu}{2ac^2} = 0 \quad (44.8)$$

44.2.8 Transmitter Clock Error

This section identifies the term $(\tau^2 - T^2)$ for the range representation defined by Eq. 44.1.

Each NAVSTAR Space Vehicle (SV) keeps its own SV Time with its own SV clock. SV Time is a proper time. According to ICD-GPS-200C, paragraph 20.3.3.3.3.1, page 88, the user transforms SV Time to GPS Time² t using equations given there, together with clock values given in each HANU message. GPS Time t is a coordinate time. Given the evaluation of ICD Eq. (1):

$$t = t_{SV} - \Delta t \quad (44.9)$$

and ICD Eq. (2):

$$\Delta t_{SV} = a_{f0} + a_{f1}(t - t_{0c}) + a_{f2}(t - t_{0c})^2 + \Delta t_r \quad (44.10)$$

then time of radio wave front emission from the NAVSTAR SV in coordinate time is t (GPS Time). There are two significant facts to note here: (i) SV Time is a NAVSTAR clock proper time, and (ii) GPS Time is a relativistic coordinate time. Note that there are no leap seconds in GPS Time. As noted in the ICD, ICD Eqs. (1) and (2) are coupled: Approximate t by t_{SV} in ICD Eq. (2). ICD-GPS-200C, presents three relativistic expressions:

$$\Delta t_r = Fe \left(A^{1/2} \right) \sin E_k \quad (44.11)$$

$$F = \frac{-2\mu^{1/2}}{c^2} \quad (44.12)$$

$$\Delta t_r = -\frac{2\vec{R} \cdot \vec{V}}{c^2} \quad (44.13)$$

The right-hand side of Eq. 44.11, using Eq. 44.12, is found in the derivation of Eqs. 44.30 and 44.32. The right-hand side of Eq. 44.13, is found in the derivation of Eqs. 44.28 and 44.29. Thus the use of Δt_r converts SV Time (a proper time) to GPS Time (a coordinate time). Therefore use Eq. 44.29 or Eq. 44.31 to explain the conversion of NAVSTAR proper time to NAVSTAR coordinate time. It is clear that the ICD expressions must be applied twice: at t_k and at t_{k+1} . ICD-GPS-208, Section 20.1, presents a related discussion for the HANU.

The frequency of each NAVSTAR SV clock is physically offset to negate the effect of the rate $3\mu/(2ac^2)$ in the term $(\tau_{k+1} - \tau_k)(3\mu/(2ac^2))$ in Eqs. 44.29 and 44.31. ICD-GPS-200C, Section 20.3.3.3.3.1, page 11, states that the NAVSTAR transmitted frequency is biased (for general relativity) by $\Delta f/f = -4.464 \times 10^{-10}$ with $\Delta f = -4.5674 \times 10^{-3} Hz$ so that the nominal frequency to a user on the ground is $10.23 MHz$. Note: Each NAVSTAR orbit has $a_{NAVSTAR} = 4.2er$. If the bias is designed for the USER at $a_{USER} = 1.0er$,

²The ICD uses t for GPS Time. I use t for TDT Time. Distinguish them by context.

there is an obvious error for a USER orbit with $a > 1.0er$. But this will perfectly alias into the phase rate estimate.

44.3 Schwarzschild Metric

Einstein's gravitational field equation (1916):

$$\mathbf{G}_{\mu\nu} = 8\pi\mathbf{T}_{\mu\nu}$$

with second order tensors $\mathbf{G}_{\mu\nu}$ and $\mathbf{T}_{\mu\nu}$, is a simple and beautiful expression to physicists, but is difficult to apply to solutions of real problems – by mathematicians and engineers. Time and space are coupled on a four-dimensional hypersurface (manifold). After selection of coordinates, this equation generates a second order nonlinear partial differential equation in four dimensions. To apply this equation, one must find an appropriate metric that is consistent with it. Two exact metric solutions (line elements) have been found, one by Karl Schwarzschild (1916) and one by Roy Kerr (1963), and a host of approximate solutions, referred to as Post-Newtonian approximations, are in use. Each metric is a scalar nonlinear ordinary first order differential equation. The Schwarzschild metric refers to a static non-rotating mass distributed in spherical concentric layers³. The Kerr metric refers to a rotating mass, and models its quadrupole (J_2). There exists a small group of relativists who have been seeking other exact solutions for many years. The Schwarzschild metric is sufficient (TBR) for our modeling of GPS clocks.

The Schwarzschild metric can be found multiple times with a WEB search, and is given by several authors of books on general relativity; *e.g.*, Moller[84], Moller Eqs 11.83 and 11.88. The Schwarzschild metric can be written in several forms, one of which is:

$$ds^2 = \left[1 - \frac{2\mu}{c^2 r}\right] c^2 dt^2 - \left[1 - \frac{2\mu}{c^2 r}\right]^{-1} dr^2 - r^2 (d\theta^2 + \sin^2 \theta d\varphi^2) \quad (44.14)$$

where:

c	speed of light
μ	gravitational constant
ds	interval in 4d (like distance in 3d)
dt	coordinate time interval
$d\tau$	proper time interval
r	radial distance from Earth CM to clock
θ	codeclination of clock
φ	right ascension of clock

$$ds = cd\tau \quad (44.15)$$

³This is a crude model for our earth.

and the 3×1 matrix of inertial Earth centered position components for the clock are defined:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} r \sin \theta \cos \varphi \\ r \sin \theta \sin \varphi \\ r \cos \theta \end{bmatrix} \quad (44.16)$$

44.3.1 Proper Time vs Coordinate Time

Define clock speed squared with \dot{s}^2 :

$$\dot{s}^2 = \sum_{j=1}^3 \dot{x}_j^2 \quad (44.17)$$

to find:

$$\dot{s}^2 = \dot{r}^2 + r^2 \left(\dot{\theta}^2 + \dot{\varphi}^2 \sin^2 \theta \right) \quad (44.18)$$

Expand $\left[1 - 2\mu/(c^2 r)\right]^{-1}$, retain terms to $O(c^{-2})$, and insert into Eq. 44.14:

$$ds = cdt \left[1 - \frac{1}{c^2} \left(\frac{2\mu}{r} + \dot{r}^2 + r^2 \left(\dot{\theta}^2 + \dot{\varphi}^2 \sin^2 \theta \right) \right) \right]^{1/2} \quad (44.19)$$

Insert Eqs. 44.15 and 44.18 into Eq. 44.19:

$$d\tau = dt \sqrt{1 - \frac{1}{c^2} \left(\frac{2\mu}{r} + \dot{s}^2 \right)} \quad (44.20)$$

Expand the radical and truncate to order $O(c^{-4})$ to get:

$$d\tau = dt \left(1 - \frac{1}{c^2} \left(\frac{\mu}{r} + \frac{\dot{s}^2}{2} \right) \right) \quad (44.21)$$

$$dt = d\tau \left(1 + \frac{1}{c^2} \left(\frac{\mu}{r} + \frac{\dot{s}^2}{2} \right) \right) \quad (44.22)$$

Potential energy μ/r captures general relativity, and kinetic energy $\dot{s}^2/2$ captures special relativity, according to the Schwarzschild metric.

44.3.2 Time Integrals

Definite time integrals of Eqs. 44.21 and 44.22 provide transformations between proper time τ and coordinate time t . Our use of t_k and t_{k+1} refers to our ODTK filter time-update function. For implementation, we require state estimate values at both t_k and t_{k+1} , in coordinate time TDT. Thus:

$$(\tau_{k+1} - \tau_k) = (t_{k+1} - t_k) \left[1 - \frac{3\mu}{2ac^2} \right] - \frac{2e\sqrt{\mu a}}{c^2} (\sin E_{k+1} - \sin E_k) \quad (44.23)$$

$$(t_{k+1} - t_k) = (\tau_{k+1} - \tau_k) \left[1 + \frac{3\mu}{2ac^2} \right] + \frac{2e\sqrt{\mu a}}{c^2} (\sin E_{k+1} - \sin E_k) \quad (44.24)$$

where classical expressions from two-body astrodynamics have been used:

$$r = a(1 - e \cos E)$$

$$M = M_0 + n(t - t_0) = E - e \sin E$$

$$\dot{M} = n = \dot{E}(1 - e \cos E)$$

$$dt = \left[\frac{a^{3/2}}{\sqrt{\mu}} \right] (1 - e \cos E) dE$$

$$n^2 a^3 = \mu$$

Since:

$$\sqrt{\mu a} e \sin E = r \dot{r} = \mathbf{r} \cdot \dot{\mathbf{r}} \quad (44.25)$$

then equivalently:

$$(\tau_{k+1} - \tau_k) = (t_{k+1} - t_k) \left[1 - \frac{3\mu}{2ac^2} \right] - \frac{2}{c^2} (r_{k+1} \dot{r}_{k+1} - r_k \dot{r}_k) \quad (44.26)$$

$$(t_{k+1} - t_k) = (\tau_{k+1} - \tau_k) \left[1 + \frac{3\mu}{2ac^2} \right] + \frac{2}{c^2} (r_{k+1} \dot{r}_{k+1} - r_k \dot{r}_k) \quad (44.27)$$

Rearrange the last two equations:

$$(\tau_{k+1} - t_{k+1}) = (\tau_k - t_k) - (t_{k+1} - t_k) \frac{3\mu}{2ac^2} - \frac{2}{c^2} (r_{k+1} \dot{r}_{k+1} - r_k \dot{r}_k) \quad (44.28)$$

$$(t_{k+1} - \tau_{k+1}) = (t_k - \tau_k) + (\tau_{k+1} - \tau_k) \frac{3\mu}{2ac^2} + \frac{2}{c^2} (r_{k+1} \dot{r}_{k+1} - r_k \dot{r}_k) \quad (44.29)$$

or, using Eq. 44.25 again:

$$(\tau_{k+1} - t_{k+1}) = (\tau_k - t_k) - (t_{k+1} - t_k) \frac{3\mu}{2ac^2} - \frac{2e\sqrt{\mu a}}{c^2} (\sin E_{k+1} - \sin E_k) \quad (44.30)$$

$$(t_{k+1} - \tau_{k+1}) = (t_k - \tau_k) + (\tau_{k+1} - \tau_k) \frac{3\mu}{2ac^2} + \frac{2e\sqrt{\mu a}}{c^2} (\sin E_{k+1} - \sin E_k) \quad (44.31)$$

Add $(t_{k+1} - \tau_{k+1})$, according to Eqs. 44.29 or 44.31, to proper time τ_{k+1} to get coordinate time t_{k+1} . Add $(\tau_{k+1} - t_{k+1})$, according to Eqs. 44.28 or 44.30, to coordinate time t_{k+1} to get proper time τ_{k+1} .

Run a minus sign through Eq. 44.28:

$$(t_{k+1} - \tau_{k+1}) = (t_k - \tau_k) + (t_{k+1} - t_k) \frac{3\mu}{2ac^2} + \frac{2}{c^2} (r_{k+1}\dot{r}_{k+1} - r_k\dot{r}_k) \quad (44.32)$$

and compare Eq. 44.32 to Eq. 44.29. They are different; *i.e.*, we have implicitly assumed that $(t_{k+1} - t_k) \frac{3\mu}{2ac^2}$ is negligibly different than $(\tau_{k+1} - \tau_k) \frac{3\mu}{2ac^2}$. This is due to the above truncation to order $O(c^{-4})$ (TBR).

44.3.3 JPL Model

Comparison of the equations above to those of JPL[132] enables identification of terms for a USER ground clock in the JPL document, inappropriate for a USER spacecraft clock in free-fall.

44.4 Group Delay Differential

From paragraph 3.3.1.7.2 of the ICD-GPS-200C[41], there exists a NAVSTAR dependent group delay differential between the radiated L_1 and L_2 signals whose magnitude can be as large as $15ns$; *i.e.*, $15ft$. From paragraph 20.3.3.3.1.8: "Bits 17 through 24 of word seven contain the L1-L2 correction term, T_{GD} , for the benefit of "L1 only" or "L2 only" users. The related user algorithm is given in paragraph 20.3.3.3.

The NAVSTAR dependent group delay differential between the radiated L_1 and L_2 signals should be applied for both cases; *i.e.*, for users of both L1 and L2, and for the benefit of "L1 only" or "L2 only" users (TBR).

44.5 Remove Ionospheric Range

Let $\rho(t)$ denote GPS pseudo-range at time-tag t , free of (first-order) ionospheric delay. Given P-Code pseudo-range values $\rho_1(t)$ and $\rho_2(t)$ on L_1 and L_2 , then according to Section 20.3.3.3.3.3 of ICD-GPS-200C[41]:

$$\rho(t) = \frac{\rho_2(t) - \gamma\rho_1(t)}{1 - \gamma} \quad (44.33)$$

where:

$$\gamma = (f_1/f_2)^2 \quad (44.34)$$

and f_1 and f_2 are carrier frequencies for L_1 and L_2 .

ICD-GPS-200C is short on definition. What is meant by $\rho(t)$, free of first-order ionospheric delay? Let i denote frequency for L_i , where $i \in \{1, 2\}$. Then for $\rho_1(t)$ and $\rho_2(t)$ on L_1 and L_2 :

$$\rho_i(t) = \rho^R(t) + \rho_i^I(t) + \rho^O(t) \quad (44.35)$$

where $\rho^R(t)$ is due to relativistic time delay, $\rho_i^I(t)$ is the frequency dependent ionospheric range, $\rho^O(t)$ is due to all range components other than $\rho^R(t)$ and $\rho_i^I(t)$, and:

$$\rho_i^I(t) = \frac{aN_T}{f_i^2}, \quad \text{where } a \text{ is a constant} \quad (44.36)$$

$$N_T = TEC, \quad \text{Total Electron Content} \quad (44.37)$$

Then by definition:

$$\rho(t) = \rho^R(t) + \rho^O(t) \quad (44.38)$$

44.5.1 Proof

ICD-GPS-200 does not provide a proof, so here is mine. Start with the numerator on the right-hand side of Eq. 44.33:

$$\begin{aligned} \rho_2(t) - \gamma\rho_1(t) &= [\rho^R(t) + \rho_2^I(t) + \rho^O(t)] - \left(\frac{f_1^2}{f_2^2}\right) [\rho^R(t) + \rho_1^I(t) + \rho^O(t)] \\ &= [\rho^R(t) + \rho^O(t)](1 - \gamma) + \frac{aN_T}{f_2^2} - \frac{aN_T}{f_2^2} \end{aligned} \quad (44.39)$$

So:

$$\frac{\rho_2(t) - \gamma\rho_1(t)}{(1 - \gamma)} = \rho^R(t) + \rho^O(t) = \rho(t) \quad (44.40)$$

satisfies the defining Eq. 44.38.

44.5.2 Phase Count

Run a differential operator through Eq. 44.33 to get:

$$\Delta\rho(t) = \frac{\Delta\rho_2(t) - \gamma\Delta\rho_1(t)}{1 - \gamma} \quad (44.41)$$

and interpret $\Delta\rho_1(t)$, $\Delta\rho_2(t)$, and $\Delta\rho(t)$ as range differences. If ionospheric range differences can be related linearly to phase count, then Eq. 44.41 provides a convenient mechanism to convert values of frequency dependent phase counts to a phase count with first order ionosphere removed.

44.6 GPS Range First-Differences

44.6.1 Notation

Symbol	Description
i	NAVSTAR SV subscript index: $i \in \{1, 2, \dots, 27\}$
u	USER spacecraft subscript identifier
t	coordinate time (TDT)
t_i	coordinate time (TDT) of radio wave front transmission from NAVSTAR i
\hat{t}_i	erroneous estimate of t_i by NAVSTAR SV clock
t_u	coordinate time (TDT) of radio wave front reception at USER user u
\hat{t}_u	erroneous estimate of t_u by USER clock
c	speed of light in vacua
$\rho_{ui}(t)$	one-way GPS range representation: NAVSTAR to USER
$\rho_{ui}^R(t)$	relativistic range (geometric range): NAVSTAR to USER
$\rho_i^X(t)$	range variation incurred due to orbit propagation error for NAVSTAR i
$\rho_{ui}^I(t)$	ionospheric range: NAVSTAR to USER
$\Delta t_i(t)$	NAVSTAR SV clock error
$\Delta t_u(t)$	USER clock error
$w_u(t)$	sum of other USER dependent range errors
$w_i(t)$	sum of other NAVSTAR SV dependent range errors
$\hat{Z}(t)$	6x1 USER orbit estimate
$\Delta \rho_{ij}(t)$	range difference between ranges from NAVSTAR SVs i and j at time t

Table 44.1: Notation

44.6.2 C/A Code Range Representation

The one-way GPS C/A code range representation $\rho_{ui}(t)$ from NAVSTAR transmitting antenna phase center to USER receiving antenna phase center is defined here by:

$$\rho_{ui}(t) = \rho_{ui}^R(t) + \rho_i^X(t) + \rho_{ui}^I(t) + c\Delta t_u(t) - c\Delta t_i(t) + w_u(t) - w_i(t) \quad (44.42)$$

where:

$$\Delta t_i(t) = t_i - \hat{t}_i$$

$$\Delta t_u(t) = t_u - \hat{t}_u$$

44.6.3 First Difference

Assume the USER spacecraft uses a JPL designed GPS BlackJack receiver. The GPS BlackJack receiver clock is autonomously steered, using BlackJack

on-board navigation solutions, to within $1\mu s$ ($300m$) (Montenbruck[86]) of GPS time. GPS time is also a coordinate time, distinct from TDT. Apparently, this clock steering is never turned off. Woodburn has demonstrated that the BlackJack clock steering errors are eliminated by forming range difference measurements for distinct NAVSTAR SVs from the C/A code range measurements. Define the difference $\Delta\rho_{ij}(t)$ for distinct SVs i and j at time t :

$$\Delta\rho_{ij}(t) = \rho_{ui}(t) - \rho_{uj}(t) \quad (44.43)$$

Apply Eq. 44.42 to get:

$$\begin{aligned} \Delta\rho_{ij}(t) = & [\rho_{ui}^R(t) - \rho_{uj}^R(t)] + [\rho_i^X(t) - \rho_j^X(t)] + [\rho_{ui}^I(t) - \rho_{uj}^I(t)] \\ & - c[\Delta t_i(t) - \Delta t_j(t)] - [w_i(t) - w_j(t)] \end{aligned} \quad (44.44)$$

where the USER clock error $c\Delta t_u(t)$, and sum of other USER-dependent range errors $w_u(t)$, have been differenced out. The term $[\rho_{ui}^R(t) - \rho_{uj}^R(t)]$ contains the orbit information we wish to extract from the range measurement difference to calculate the USER orbit estimate $\hat{Z}(t)$.

The other terms in Eq. 44.44 will degrade the USER orbit estimate $\hat{Z}(t)$ if the state estimate has no parameters to absorb them. The term $[\rho_i^X(t) - \rho_j^X(t)]$ is due to NAVSTAR orbit estimation and propagation errors for SVs i and j , the term $[\rho_{ui}^I(t) - \rho_{uj}^I(t)]$ is due to ionospheric perturbation to the range measurements, the term $c[\Delta t_i(t) - \Delta t_j(t)]$ is due to NAVSTAR clock estimation and propagation errors for SVs i and j , and the term $[w_i(t) - w_j(t)]$ is due to other NAVSTAR-dependent effects for SVs i and j , or NAVSTAR/USER-dependent effects.

44.7 Partial Derivatives for HANU

44.7.1 ICD-GPS-208 18 April 1983

Refer to ICD-GPS-208 18 April 1983[42].

Let us denote the 8x8 HANU covariance as P , and the 8x8 uppertriangular ECSR as R . According to page I-10 Table 10.1-6, R provides covariance square root position and velocity in RAC components in units meters and seconds, and then covariance square root clock bias (sec) and drift (sec/sec (unitless)), where:

$$P = RR^T \quad (44.45)$$

44.7.2 Partial Derivatives

Differentiate Eq. 44.43:

$$\frac{\partial \Delta\rho_{ij}(t)}{\partial Z(t)} = \frac{\partial \rho_{ui}(t)}{\partial Z(t)} - \frac{\partial \rho_{uj}(t)}{\partial Z(t)} \quad (44.46)$$

The right-hand terms are given elsewhere.

44.7.3 Map HANU Covariance P to GPS Measurement Covariance P_y

Let Δy denote a 2x1 matrix on USER-NAVSTAR pseudo-range and pseudo-range-rate measurement errors. Let P_y denote a 2x2 covariance matrix on Δy . Let ΔX denote a 3x1 matrix for NAVSTAR position error in inertial components, and let $\Delta \dot{X}$ denote a 3x1 matrix for NAVSTAR velocity error in inertial components. Let $\Delta \tau$ denote NAVSTAR clock phase error, and let $\Delta \dot{\tau}$ denote NAVSTAR clock drift error. And let ΔH denote an 8x1 matrix for the NAVSTAR orbit and clock error:

$$\Delta H = \begin{bmatrix} \Delta X \\ \Delta \dot{X} \\ \Delta \tau \\ \Delta \dot{\tau} \end{bmatrix} \quad (44.47)$$

Let $\partial y / \partial H$ denote a 2x8 Jacobian matrix on partial derivatives to map NAVSTAR orbit and clock error ΔH to pseudo-range and pseudo-range-rate measurement error Δy :

$$\Delta y = \begin{bmatrix} \partial y \\ \partial H \end{bmatrix} \Delta H \quad (44.48)$$

Then:

$$P_y = \begin{bmatrix} \partial y \\ \partial H \end{bmatrix} P \begin{bmatrix} \partial y \\ \partial H \end{bmatrix}^T \quad (44.49)$$

where:

$$\begin{bmatrix} \partial y \\ \partial H \end{bmatrix} = \begin{bmatrix} \frac{\partial \rho}{\partial X} & \frac{\partial \rho}{\partial \dot{X}} & \frac{\partial \rho}{\partial \tau} & \frac{\partial \rho}{\partial \dot{\tau}} \\ \frac{\partial \dot{\rho}}{\partial X} & \frac{\partial \dot{\rho}}{\partial \dot{X}} & \frac{\partial \dot{\rho}}{\partial \tau} & \frac{\partial \dot{\rho}}{\partial \dot{\tau}} \end{bmatrix} \quad (44.50)$$

Calculate the partials in $\partial y / \partial H$ according to Eqs. 44.89 through 44.88.

Instantaneous vector relations are sufficient for analytic measurement partial derivatives.

Let \mathbf{r} denote the USER spacecraft position vector with 3x1 matrix of inertial components x , \mathbf{R} denote the NAVSTAR position vector with 3x1 matrix of inertial components X , and $\boldsymbol{\rho}$ denote the NAVSTAR to USER range vector, all at the same time t . Define:

$$\mathbf{r} = \mathbf{R} + \boldsymbol{\rho} \quad (44.51)$$

in agreement with Fig. 44.1. Differentiate with time to get inertial velocity $\dot{\mathbf{r}}$:

$$\dot{\mathbf{r}} = \dot{\mathbf{R}} + \dot{\boldsymbol{\rho}} \quad (44.52)$$

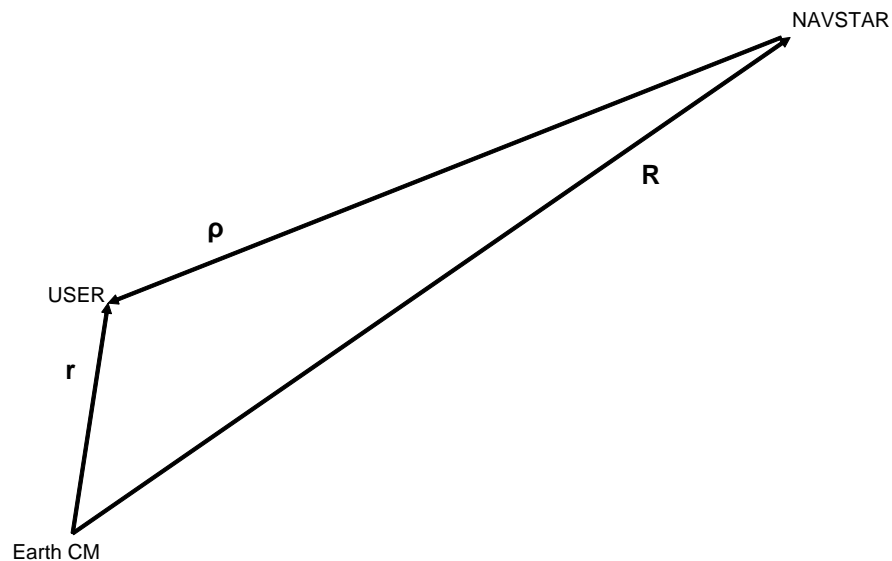


Figure 44.1: Earth-USER-NAVSTAR Vector Triangle

44.7.4 Initial Differential Equations for Variations in Range and Range-Rate

Take total differentials of Eqs. 44.51 and 44.52 (virtual displacements (La-Grange)) to get:

$$\Delta \mathbf{r} = \Delta \mathbf{R} + \Delta \boldsymbol{\rho} \quad (44.53)$$

$$\Delta \dot{\mathbf{r}} = \Delta \dot{\mathbf{R}} + \Delta \dot{\boldsymbol{\rho}} \quad (44.54)$$

Rearrange equations:

$$\boldsymbol{\rho} = \mathbf{r} - \mathbf{R} \quad (44.55)$$

$$\dot{\boldsymbol{\rho}} = \dot{\mathbf{r}} - \dot{\mathbf{R}} \quad (44.56)$$

$$\Delta \boldsymbol{\rho} = \Delta \mathbf{r} - \Delta \mathbf{R} \quad (44.57)$$

$$\Delta\dot{\rho} = \Delta\dot{\mathbf{r}} - \Delta\dot{\mathbf{R}} \quad (44.58)$$

Define the instantaneous range ρ (length of vector $\boldsymbol{\rho}$) with:

$$\rho = \sqrt{\boldsymbol{\rho} \cdot \boldsymbol{\rho}} \quad (44.59)$$

Define the instantaneous unit range vector \mathbf{L} with:

$$\boldsymbol{\rho} = \rho\mathbf{L} \quad (44.60)$$

Differentiate with time to get:

$$\dot{\boldsymbol{\rho}} = \dot{\rho}\mathbf{L} + \rho\dot{\mathbf{L}} \quad (44.61)$$

Also, take total differentials to get:

$$\Delta\boldsymbol{\rho} = \Delta\rho\mathbf{L} + \rho\Delta\mathbf{L} \quad (44.62)$$

where $\dot{\rho}$ is the instantaneous range-rate. Then:

$$\rho = \mathbf{L} \cdot \boldsymbol{\rho} \quad (44.63)$$

$$\dot{\rho} = \mathbf{L} \cdot \dot{\boldsymbol{\rho}} \quad (44.64)$$

because $\mathbf{L} \cdot \mathbf{L} = 1$ and $\mathbf{L} \cdot \dot{\mathbf{L}} = 0$. Also:

$$\rho\dot{\rho} = \boldsymbol{\rho} \cdot \dot{\boldsymbol{\rho}} \quad (44.65)$$

Take total differentials of Eq. 44.65 for:

$$\Delta\rho\dot{\rho} + \rho\Delta\dot{\rho} = \Delta\boldsymbol{\rho} \cdot \dot{\boldsymbol{\rho}} + \boldsymbol{\rho} \cdot \Delta\dot{\boldsymbol{\rho}}$$

from which:

$$\rho\Delta\dot{\rho} = \dot{\boldsymbol{\rho}} \cdot \Delta\boldsymbol{\rho} + \boldsymbol{\rho} \cdot \Delta\dot{\boldsymbol{\rho}} - \Delta\rho\dot{\rho} \quad (44.66)$$

Take total differentials of Eq. 44.63 to get:

$$\Delta\rho = \mathbf{L} \cdot \Delta\boldsymbol{\rho} \quad (44.67)$$

because $\Delta\mathbf{L} \cdot (\rho\mathbf{L}) = 0$. Insert Eq. 44.57 into Eq. 44.67 to get:

$$\Delta\rho = \mathbf{L} \cdot \Delta\mathbf{r} - \mathbf{L} \cdot \Delta\mathbf{R} \quad (44.68)$$

Insert Eqs. 44.68, 44.57, and 44.58 into Eq. 44.66 to show:

$$\begin{aligned} \rho\Delta\dot{\rho} &= \dot{\boldsymbol{\rho}} \cdot (\Delta\mathbf{r} - \Delta\mathbf{R}) - \dot{\rho}(\mathbf{L} \cdot \Delta\mathbf{r} - \mathbf{L} \cdot \Delta\mathbf{R}) + \boldsymbol{\rho} \cdot (\Delta\dot{\mathbf{r}} - \Delta\dot{\mathbf{R}}) \\ &= (\dot{\boldsymbol{\rho}} - \dot{\rho}\mathbf{L}) \cdot \Delta\mathbf{r} + \boldsymbol{\rho} \cdot \Delta\dot{\mathbf{r}} - (\dot{\boldsymbol{\rho}} - \dot{\rho}\mathbf{L}) \cdot \Delta\mathbf{R} - \boldsymbol{\rho} \cdot \Delta\dot{\mathbf{R}} \end{aligned}$$

from which:

$$\Delta\dot{\rho} = \tilde{\rho} \cdot \Delta\mathbf{r} + \mathbf{L} \cdot \Delta\dot{\mathbf{r}} - \tilde{\rho} \cdot \Delta\mathbf{R} - \mathbf{L} \cdot \Delta\dot{\mathbf{R}} \quad (44.69)$$

where:

$$\tilde{\rho} = (\dot{\rho} - \dot{\rho}\mathbf{L}) / \rho \quad (44.70)$$

Eqs. 44.68 and 44.69 are my initial vector differential equations for variations in instantaneous range and range-rate.

44.7.5 Position and Velocity Variations in Inertial Component Matrices

Let x and X denote the 3×1 matrices of inertial components of the USER position vector \mathbf{r} and NAVSTAR position vector \mathbf{R} . Let $[\mathbf{i}]$ denote the conventional 3×1 matrix of orthonormal inertial vectors. Then:

$$\mathbf{L} = L^T [\mathbf{i}] \quad (44.71)$$

$$\tilde{\rho} = \tilde{\rho}^T [\mathbf{i}] \quad (44.72)$$

$$\Delta\mathbf{r} = [\mathbf{i}]^T \Delta x \quad (44.73)$$

$$\Delta\dot{\mathbf{r}} = [\mathbf{i}]^T \Delta\dot{x} \quad (44.74)$$

$$\Delta\mathbf{R} = [\mathbf{i}]^T \Delta X \quad (44.75)$$

$$\Delta\dot{\mathbf{R}} = [\mathbf{i}]^T \Delta\dot{X} \quad (44.76)$$

44.7.6 Partialials

Insert Eqs. 44.73, 44.74, 44.75, 44.76, 44.71, and 44.72 into Eqs. 44.68 and 44.69 to get:

$$\Delta\rho = [L^T] \Delta x + [-L^T] \Delta X \quad (44.77)$$

$$\Delta\dot{\rho} = [\tilde{\rho}^T] \Delta x + [L^T] \Delta\dot{x} + [-\tilde{\rho}^T] \Delta X + [-L^T] \Delta\dot{X} \quad (44.78)$$

The above two expressions are complete due to use of total differentials. Formally:

$$\Delta\rho = \left[\frac{\partial\rho}{\partial x} \right] \Delta x + \left[\frac{\partial\rho}{\partial\dot{x}} \right] \Delta\dot{x} + \left[\frac{\partial\rho}{\partial X} \right] \Delta X + \left[\frac{\partial\rho}{\partial\dot{X}} \right] \Delta\dot{X} \quad (44.79)$$

$$\Delta\dot{\rho} = \left[\frac{\partial\dot{\rho}}{\partial x} \right] \Delta x + \left[\frac{\partial\dot{\rho}}{\partial \dot{x}} \right] \Delta \dot{x} + \left[\frac{\partial\dot{\rho}}{\partial X} \right] \Delta X + \left[\frac{\partial\dot{\rho}}{\partial \dot{X}} \right] \Delta \dot{X} \quad (44.80)$$

Compare terms to derive the desired 1x3 matrices of partial derivative Jacobians:

$$\frac{\partial\rho}{\partial x} = L^T, \quad \text{range with respect to USER position} \quad (44.81)$$

$$\frac{\partial\rho}{\partial \dot{x}} = 0_{1 \times 3}, \quad \text{range with respect to USER velocity} \quad (44.82)$$

$$\frac{\partial\dot{\rho}}{\partial x} = \tilde{\rho}^T, \quad \text{range-rate with respect to USER position} \quad (44.83)$$

$$\frac{\partial\dot{\rho}}{\partial \dot{x}} = L^T, \quad \text{range-rate with respect to USER velocity} \quad (44.84)$$

$$\frac{\partial\rho}{\partial\tau} = c, \quad \text{range with respect to USER clock phase (light speed)} \quad (44.85)$$

$$\frac{\partial\rho}{\partial\dot{\tau}} = 0, \quad \text{range with respect to USER clock drift} \quad (44.86)$$

$$\frac{\partial\dot{\rho}}{\partial\tau} = 0, \quad \text{range-rate with respect to USER clock phase} \quad (44.87)$$

$$\frac{\partial\dot{\rho}}{\partial\dot{\tau}} = 1, \quad \text{range-rate with respect to USER clock drift} \quad (44.88)$$

$$\frac{\partial\rho}{\partial X} = -L^T, \quad \text{range with respect to NAVSTAR position} \quad (44.89)$$

$$\frac{\partial\rho}{\partial \dot{X}} = 0_{1 \times 3}, \quad \text{range with respect to NAVSTAR velocity} \quad (44.90)$$

$$\frac{\partial\dot{\rho}}{\partial X} = -\tilde{\rho}^T, \quad \text{range-rate with respect to NAVSTAR position} \quad (44.91)$$

$$\frac{\partial\dot{\rho}}{\partial \dot{X}} = -L^T, \quad \text{range-rate with respect to NAVSTAR velocity} \quad (44.92)$$

where $\tilde{\rho}^T$ is the 1x3 row matrix of inertial components of vector $\tilde{\rho} = (\dot{\rho} - \dot{\rho}\mathbf{L}) / \rho$. The subset of partials referred to the USER position and velocity may be checked against those given by Koskela (page 138). The NAVSTAR partials could have been derived from the USER partials by skew-symmetry. But we feel more secure to explicitly derive them all.

Chapter 45

GPS Carrier Phase Count

45.1 Introduction

My purpose here is to derive and present:

- a representation for the GPS Doppler phase count measurement for a *generic* USER spacecraft GPS receiver
- instructions for implementation of GPS Doppler phase count measurements
- state parameters to be estimated
- expressions for required partial derivatives

Optimal orbit determination requires the identification and estimation of all observable unknown serially-correlated stochastic variables. The USER spacecraft GPS receiver clock produces an observable unknown serially-correlated stochastic variable; viz, the frequency random walk variable identified by Allan[1]. Eqs. 45.11, 45.12, and 45.13 explicitly define this variable δf_n^U , and its relation to the GPS Doppler phase count measurement representation. An explicit representation of δf_n^U is required for phase count measurement *simulation*, for sequential filter design, for construction and analysis of a measurement first difference model, and for the associated optimal orbit determination from real data.

Eqs. 45.72 and 45.73, together with the sequence of bullets that follow, present the frequency dependent Doppler phase count measurement representation N_j^{ni} for a particular USER-NAVSTAR pair. These equations are inclusive of an ionosphere model, rigorously derived, that requires the evaluation of TEC at beginning and end of the phase count time interval.

Eq. 45.86 defines construction of the ionosphere-free measurement N_j^{Ei} , and its proof follows immediately. The single difference ΔN_j^E defined by Eq. 45.118 is shown to cancel all USER clock effects.

45.1.1 BlackJack GPS Receiver

Cal Tech's BlackJack GPS receiver¹ recovers a modified form of both L1 P-code (P1) and L2 P-code (P2) ranging², in addition to C/A ranging, and provides carrier phase-count measurements on L1, L2, and C/A. According to Kerric Hill (Bruce Haines (JPL)):

- The BlackJack clock frequency is steered so that its phase is always within $1\mu s$ of GPS time.
- The BlackJack GPS receiver does not provide P1 and P2 pseudo-range measurements directly, but rather, provides measurements that have been "smoothed" using the phase count measurements.

We are not in possession of the BlackJack clock frequency steering algorithm. Unfortunately, all measurements are corrupted by the frequency steering. Constructed first difference measurements, referred to distinct NAVSTARs, are required to remove the effects of USER BlackJack clock frequency steering on both range and phase count measurements.

The BlackJack pseudo-range measurements are significantly improved due to phase count measurement "polynomial smoothing"; *i.e.*, the BlackJack P1 and P2 pseudo-range measurements are superpositions of pseudo-range and phase count Doppler. But this generates a problem for those who would process both sets of measurements: Pseudo-range "measurements" are correlated with the phase count measurements, and this destroys the ability to model their statistics appropriately. A *Hofmann-Wellenhof et. al.* smoothing algorithm is presented at http://www.gmat.unsw.edu.au/snap/gps/gps_survey/chap6/6411.htm. We cannot confirm that this is the BlackJack algorithm.

45.2 Implementation

45.2.1 Estimation of Clock Parameters

The sampled frequency sequence of every clock contains both additive white noise and additive random walk sequences. The generalized integrals I_j^{Ni} and I_j^{Uj} defined by Eqs. 45.27 and 45.28 contain both integral white noise (random walk) and integral random walk sequences. Thus the sampled phase sequence of every clock contains both random walk and integral random walk sequences. In addition, the sampled phase sequence of every clock contains a flicker noise sequence. Flicker noise is important for NAVSTAR atomic clocks, but is relatively unimportant for USER crystal clocks.

An option is required to estimate two clock parameters, phase and frequency, for each NAVSTAR clock with every GPS measurement combination. An option is required to estimate two clock parameters, phase and frequency, for each

¹BlackJack is available commercially from Spectrum Astro.

²US Patent 6,061,390, CIT, Meehan, Thomas, Brooks, Young, and Lawrence, May 9, 2000.

USER spacecraft when single measurement differencing is not used. The existing stochastic crystal clock model is to be used for both NAVSTAR clocks and USER clocks. A flicker noise model will be added later for NAVSTAR atomic clocks. Estimate random walk and integral random walk sequences in phase according to Eqs. 45.172 through 45.175 in the filter implementation.

45.2.2 L_1 Phase Count Measurements

Eq. 45.73 presents the frequency-dependent NAVSTAR-dependent Doppler phase count representation N_j^{ni} in final form, at time tag t_j .

45.2.3 L_1, L_2 Phase Count Measurements

Remove Ionosphere Perturbations

The Doppler phase count measurement representation N_j^{Ei} over $[t_j - \Delta t, t_j]$ is given by Eq. 45.86, where γ is defined by Eq. ???. This measurement representation combines the two phase-count measurements N_j^{1i} and N_j^{2i} to remove first order ionosphere effects. USER clock effects and NAVSTAR clock effects remain.

Single Differenced Phase Count Measurements

Given simultaneous phase count measurements N_j^{Ep} and N_j^{Eq} on independent receiver channels, where p and q refer to distinct NAVSTARS, they can be differenced to form $\Delta N_j^E = N_j^{Ep} - N_j^{Eq}$, according to Eq. 45.118. The derived measurement ΔN_j^E is then free of all USER clock effects, as demonstrated.

45.2.4 Partial Derivatives

See Section 45.4.7 for orbit partials, and Section 45.10 for non-orbit partials. The phase count measurement has a range difference representation, so partials of phase count with respect to position and velocity are reduced to the difference of partials of range with respect to position and velocity.

45.3 Notation

45.3.1 Time

Proper Time τ refers to the phase of some clock, or to the mean phase of an ensemble of clocks. *Coordinate Time* t is the fourth relativistic coordinate, where the first three coordinates are used to denote a clock position referred to an inertial frame. Proper time and coordinate time are related approximately with the differential equation:

$$d\tau = dt \sqrt{1 - \left[\frac{2\mu}{c^2 r} + \frac{\dot{s}^2}{c^2} \right]} \quad (45.1)$$

derived from the Schwarzschild metric, where μ refers to the two-body geocentric gravitational constant, r and \dot{s} are spacecraft position and velocity magnitudes, and c is the speed of light in vacua. *Proper Time* is associated with clocks, and *Coordinate Time* is associated with spacecraft position and velocity. Eq. 45.1 provides the bridge. Given an epoch, say t_0 , where the relation between τ_0 and t_0 is estimated or otherwise known, then Eq. 45.1 can be integrated to propagate the relation.

45.3.2 Oscillator and Clock

Every clock is an artifact with two parts, an oscillator and a counter of oscillations or phase referred an origin. Some clocks employ "gears" to transform phase count to hours, minutes, and seconds.

The TAI (International Atomic Time) second is defined: $1 \text{ sec}_{TAI} = 9,192,631,770$ cycles³ between two hyperfine levels of the ground state of cesium at geoid mean sea level. The number of seconds after any origin refers to an associated number of oscillator cycles after that origin. So time refers to the accumulated phase (number of cycles) of a clock.

The phase of one oscillator can be physically compared and differenced with that of another oscillator. Phase is physically measurable only in this sense, but frequency is not physically measureable in any sense. Frequency is derived by calculating the change in phase with *time*, where *time* is the phase of an independent clock.

45.3.3 Generalized Integral

Stochastic sequences cannot be differentiated⁴ or integrated, but they can be differenced and summed. We will refer to these differences and sums as generalized derivatives and generalized integrals⁵, or just derivatives and integrals.

45.3.4 Doppler Phase Count

The symbol N_j^{ni} denotes the Doppler phase count measurement representation:

- referred to the USER spacecraft oscillator
- over time interval $[t_j - \Delta t, t_j]$
- with time tag at t_j

³The mean value from an ensemble of oscillators is used.

⁴By definition, the derivative requires application of a limit that does not exist for stochastic sequences.

⁵My use of *generalized integral* is conventional.

- on L-BAND frequency L_n , $n \in \{1, 2\}$
- for NAVSTAR i , $i \in \{1, 2, \dots, 27\}$

N_j^{ni} is inclusive of:

- an integral on Doppler frequency
- an integral on random USER clock effects
- an integral on random NAVSTAR clock effects
- USER receiver PLL (Phase Lock Loop) thermal noise
- ionospheric effects

When there is no need to identify frequency f_n or NAVSTAR i , we shall replace N_j^{ni} with N_j .

45.3.5 Two Types of Phase Count Error

We are interested here in distinguishing phase count measurement errors according to two types.

Type 1 Error

Let us denote a Type 1 error in N_k with random variable $\delta N_k^{(1)}$, for $k \in \{1, 2, 3, \dots\}$. According to Kaplan[50], Type 1 errors are dominated by thermal noise. Then Type 1 errors are dominantly Gaussian white noise⁶, are not observable from Doppler phase count measurements, cannot be estimated, but must be accounted for statistically in the sequential filter for orbit determination.

Type 2 Error

Let us denote a Type 2 error in N_k with random variable $\delta N_k^{(2)}$, for $k \in \{1, 2, 3, \dots\}$. Let us denote USER receiver clock time with t_j , and USER receiver clock timing error with random variable δt_k , for $k \in \{1, 2, 3, \dots\}$.

Define:

$$\delta N_k^{(2)} = f_k \delta t_k \quad (45.2)$$

Both δt_k and $\delta N_k^{(2)}$ are composed of sequences in random walk and integral random walk. Hence they are serially correlated. Therefore the observable part of $\delta N_k^{(2)}$ must be estimated and removed.

⁶All thermal noise is Gaussian and white[16].

45.4 Receiver Doppler Algorithm

45.4.1 NAVSTAR Frequencies

Nominal GPS NAVSTAR L-BAND frequencies \bar{f}_n^N , $n \in \{1, 2\}$, are defined by:

$$L_1 : \bar{f}_1^N = (154) \bar{f}_0^N \quad (45.3)$$

$$L_2 : \bar{f}_2^N = (120) \bar{f}_0^N \quad (45.4)$$

where⁷:

$$\bar{f}_0^N = (10.23 \text{ MHz}) (1 - 4.4647 \times 10^{-10}) \quad (45.5)$$

The transmitter on each GPS NAVSTAR emits radio signals:

$$f_1^N = \bar{f}_1^N + \delta f_1^N = (154) f_0^N \quad (45.6)$$

$$f_2^N = \bar{f}_2^N + \delta f_2^N = (120) f_0^N \quad (45.7)$$

where:

$$f_0^N = \bar{f}_0^N + \delta f_0^N \quad (45.8)$$

$$\delta f_1^N = (154) \delta f_0^N \quad (45.9)$$

$$\delta f_2^N = (120) \delta f_0^N \quad (45.10)$$

where f_0^N is generated by the NAVSTAR oscillator, and where δf_0^N is a random perturbation on the nominal frequency \bar{f}_0^N .

45.4.2 Doppler Frequency

A USER spacecraft receives a Doppler shifted NAVSTAR signal, contaminated by additive ionospheric effects, with frequency⁸ $\beta_{ij} f_n^N + \delta f_n^{ION}$, where β_{ij} and δf_n^{ION} denote Doppler shift and ionospheric effect. This is subtracted⁹ from frequency f_n^U , generated by the USER spacecraft oscillator, to define the instantaneous Doppler frequency f_n^D :

⁷The perturbation is defined such that a ground receiver will believe that the fundamental frequency $f_0 = 10.23$ MHz.

⁸The received carrier frequency is not directly measured. Rather, the GPS receiver *measures* phase, and derives frequency as the change in phase with time, according to the USER clock. The phase *measurement* is performed using closed-loop feedback control systems, referred to as *frequency locked loop* (FLL) and *Costas Phase Locked Loop* (PLL). On signal acquisition[50], an FLL is employed with carrier tracking loop bandwidth wide, followed by a PLL with carrier tracking loop bandwidth as narrow as possible.

⁹The subtraction order is defined by each actual receiver, and so my representation here may need a sign change for some receivers.

$$f_n^D = f_n^U - (\beta_{ij} f_n^N + \delta f_n^{ION}) \quad (45.11)$$

NAVSTAR clock phenomenology is embedded in f_n^N , and USER clock phenomenology is embedded in f_n^U . USER clock phenomenology is assumed to dominate NAVSTAR clock phenomenology. Then the difference $(f_n^U - f_n^N)$, captures USER clock phenomenology most significantly:

$$\delta f_n^U = f_n^U - f_n^N \quad (45.12)$$

Solve Eq. 45.12 for f_n^U , and insert into Eq. 45.11 to get:

$$f_n^D = f_n^N (1 - \beta_{ij}) + \delta f_n^U - \delta f_n^{ION} \quad (45.13)$$

Eqs. 45.6 and 45.7 can be written:

$$f_n^N = \bar{f}_n^N + \delta f_n^N \quad (45.14)$$

where \bar{f}_n^N is known and defined by Eqs. 45.3 and 45.4. δf_n^N is due to unknown NAVSTAR clock phenomenology. Insert Eq. 45.14 into Eq. 45.13 to get:

$$f_n^D = (\bar{f}_n^N + \delta f_n^N) (1 - \beta_{ij}) + \delta f_n^U - \delta f_n^{ION} \quad (45.15)$$

Eq. 45.15 presents the USER clock phenomenology term δf_n^U unscaled by the Doppler shift β_{ij} , and the NAVSTAR clock phenomenology δf_n^N scaled by $(1 - \beta_{ij})$. The product $\delta f_n^N (1 - \beta_{ij})$ is a very complicated random function of time. The random frequency scalar δf on every clock has two significant additive components, white noise and random walk.

45.4.3 Define Phase Count

Positive zero crossings (cycles) of f_n^D are counted and accumulated (integrated) by the USER receiver, inclusive of a partial cycle, across a USER defined phase count interval¹⁰ $[t_j - \Delta t, t_j]$, to construct a rational phase count number N_j^{ni} , time-tag t_j for $j \in \{1, 2, \dots\}$, NAVSTAR $i \in \{1, 2, \dots, 27\}$. N_j^{ni} consists of the sum of an integer number of cycles and a partial cycle. The GPS receiver Costas Phase Lock Loop (PLL) incurs thermal noise[50] δN_j^{ni} , independent for each NAVSTAR link i . Thus N_j^{ni} has the representation with the generalized integral:

$$N_j^{ni} = \int_{t_j - \Delta t}^{t_j} f_n^D dt + \delta N_j^{ni} \quad (45.16)$$

When f_n^D has units cycles/s, and t has units s, then N_j^{ni} has units cycles.

¹⁰ t_j is a relativistic coordinate time; e.g., TCG or GPS Time.

Begin

Questions:

- Are thermal noise phase components δN_j^{1i} and δN_j^{2i} frequency dependent?
- Does there exist δN_j^0 in the GPS receiver such that we can write:

$$\delta N_j^{1i} = 154 \delta N_j^0$$

$$\delta N_j^{2i} = 120 \delta N_j^0$$

where in the NAVSTAR GPS transmitter we know:

$$f_1^N = (154) f_0^N$$

$$f_2^N = (120) f_0^N$$

End**45.4.4 Doppler Shift**

Consider an L-BAND wave front emitted from a NAVSTAR spacecraft at time t_i and received by the USER spacecraft at time $t_j > t_i$. The unitless Doppler shift β can be approximated¹¹ to order $O(c^{-3})$ with:

$$\beta_{ij} = 1 - \frac{1}{c} \dot{\rho}_{ij} + \frac{\mu}{c^2} \left[\frac{1}{r_j} - \frac{1}{R_i} \right] + \frac{1}{2c^2} \left[\dot{s}_j^2 - \dot{S}_i^2 \right] \quad (45.17)$$

where:

- c is speed of light in vaccua
- μ is the geocentric two-body gravitational constant
- r_j and \dot{s}_j are magnitudes of USER spacecraft position and velocity vectors at time t_j
- R_i and \dot{S}_i are magnitudes of NAVSTAR spacecraft position and velocity vectors at time t_i
- $\dot{\rho}_{ij}$ is the range-rate $\dot{\rho}_{ij} = d\rho_{ij}/dt$ at time $t = t_j$
- ρ_{ij} is the range between NAVSTAR spacecraft at time t_i and USER spacecraft at time t_j

¹¹This approximation derives from the use of the Schwarzschild metric, derived from Eq. ???. Appendix D herein provides a derivation of this expression, given the Schwarzschild metric.

Approximately:

$$\int_{t_j - \Delta t}^{t_j} \dot{\rho}_{ij} dt = \rho_{ij}(t_j) - \rho_{ij}(t_j - \Delta t) \quad (45.18)$$

Define:

$$\Delta\rho_{ij} = \rho_{ij}(t_j) - \rho_{ij}(t_j - \Delta t) \quad (45.19)$$

$$\chi_{ij} = \int_{t_j - \Delta t}^{t_j} \left\{ \frac{\mu}{c^2} \left[\frac{1}{r_j} - \frac{1}{R_i} \right] + \frac{1}{2c^2} \left[\dot{s}_j^2 - \dot{S}_i^2 \right] \right\} dt \quad (45.20)$$

$$\zeta_{ij} = \Delta\rho_{ij}/c - \chi_{ij} \quad (45.21)$$

where $\Delta\rho_{ij}$ has distance units, χ_{ij} has time units, and ζ_{ij} has time units. Then:

$$\zeta_{ij} = \int_{t_j - \Delta t}^{t_j} (1 - \beta) dt \quad (45.22)$$

Relativity

Eq. 45.17 has two relativistic terms: $\frac{\mu}{c^2} \left[\frac{1}{r_j} - \frac{1}{R_i} \right]$ is due to general relativity and $\frac{1}{2c^2} \left[\dot{s}_j^2 - \dot{S}_i^2 \right]$ is due to special relativity. This is important for one-way Doppler measurements. Position magnitudes R_i and r_j refer to NAVSTAR and USER spacecraft clock positions. Since $R_i \neq r_j$, then $\frac{\mu}{c^2} \left[\frac{1}{r_j} - \frac{1}{R_i} \right]$ must be modeled. Velocity magnitudes \dot{S}_i and \dot{s}_j refer to NAVSTAR and USER spacecraft clock inertial speeds. Since $\dot{S}_i \neq \dot{s}_j$, then $\frac{1}{2c^2} \left[\dot{s}_j^2 - \dot{S}_i^2 \right]$ must be modeled. On the other hand, if the same ground-fixed clock is used for transmitter and receiver in a two-way Doppler measurement, then $R_i = r_j$ and $\dot{S}_i = \dot{s}_j$, and relativity vanishes.

45.4.5 Phase Count Representation

Insert Eq. 45.15 into Eq. 45.16 to get:

$$N_j^{ni} = \int_{t_j - \Delta t}^{t_j} [(\bar{f}_n^N + \delta f_n^N) (1 - \beta_{ij}) + \delta f_n^U - \delta f_n^{ION}] dt + \delta N_j^{ni} \quad (45.23)$$

The nominal frequency \bar{f}_n^N is a time constant. Then:

$$N_j^{ni} = \bar{f}_n^N \int_{t_j - \Delta t}^{t_j} (1 - \beta_{ij}) dt + \int_{t_j - \Delta t}^{t_j} \delta f_n^N (1 - \beta_{ij}) dt + \int_{t_j - \Delta t}^{t_j} \delta f_n^U dt - \int_{t_j - \Delta t}^{t_j} \delta f_n^{ION} dt + \delta N_j^{ni} \quad (45.24)$$

Let us denote clock phase phenomenology of NAVSTAR and USER clocks with I_j^{nNi} and I_j^U , with time-tags t_j , where I_j^{nNi} is referred to NAVSTAR i . Define I_j^{nNi} and I_j^U with:

$$\int_{t_j-\Delta t}^{t_j} \delta f_n^N (1 - \beta_{ij}) dt = \bar{f}_n^N \int_{t_j-\Delta t}^{t_j} \left[\frac{\delta f_n^N}{\bar{f}_n^N} \right] (1 - \beta_{ij}) dt = \bar{f}_n^N I_j^{nNi} \quad (45.25)$$

$$\int_{t_j-\Delta t}^{t_j} \delta f_n^U dt = \bar{f}_n^U \int_{t_j-\Delta t}^{t_j} \left[\frac{\delta f_n^U}{\bar{f}_n^U} \right] dt = \bar{f}_n^U I_j^{nU} \quad (45.26)$$

That is:

$$I_j^{nNi} = \int_{t_j-\Delta t}^{t_j} \left[\frac{\delta f_n^N}{\bar{f}_n^N} \right] (1 - \beta_{ij}) dt \quad (45.27)$$

$$I_j^{nU} = \int_{t_j-\Delta t}^{t_j} \left[\frac{\delta f_n^U}{\bar{f}_n^U} \right] dt \quad (45.28)$$

where $\delta f_n^N / \bar{f}_n^N$ and $\delta f_n^U / \bar{f}_n^U$ are fractional frequencies on NAVSTAR and USER clocks. Eq. 45.24 becomes:

$$N_j^{ni} = \bar{f}_n^N \int_{t_j-\Delta t}^{t_j} (1 - \beta_{ij}) dt + \bar{f}_n^N I_j^{nNi} + \bar{f}_n^U I_j^{nU} - \int_{t_j-\Delta t}^{t_j} \delta f_n^{ION} dt + \delta N_j^{ni} \quad (45.29)$$

Insert Eq. 45.22 into Eq. 45.29:

$$N_j^{ni} = \bar{f}_n^N \zeta_{ij} + \bar{f}_n^N I_j^{nNi} + \bar{f}_n^U I_j^{nU} - \int_{t_j-\Delta t}^{t_j} \delta f_n^{ION} dt + \delta N_j^{ni} \quad (45.30)$$

45.4.6 Approximations for χ_{ij}

The integral for Eq. 45.20 can be approximated. Define:

$$\eta = \frac{\mu}{c^2} \left[\frac{1}{r_j} - \frac{1}{R_i} \right] + \frac{1}{2c^2} \left[\dot{s}_j^2 - \dot{S}_i^2 \right] \quad (45.31)$$

$$\eta^{USER} = \frac{\mu}{c^2 r_j} + \frac{\dot{s}_j^2}{2c^2} \quad (45.32)$$

$$\eta^{NAV} = \frac{\mu}{c^2 R_i} + \frac{\dot{S}_i^2}{2c^2} \quad (45.33)$$

Then:

$$\eta = \eta^{USER} + \eta^{NAV} \quad (45.34)$$

Define:

$$\chi_{ij}^{USER} = \int_{t_j - \Delta t}^{t_j} \left\{ \frac{\mu}{c^2 r_j} + \frac{\dot{s}_j^2}{2c^2} \right\} dt \quad (45.35)$$

$$\chi_{ij}^{NAV} = - \int_{t_j - \Delta t}^{t_j} \left\{ \frac{\mu}{c^2 R_i} + \frac{\dot{S}_i^2}{2c^2} \right\} dt \quad (45.36)$$

Then:

$$\chi_{ij} = \chi_{ij}^{USER} + \chi_{ij}^{NAV} \quad (45.37)$$

Use the symbol over-bar to denote mean values: $\bar{\eta}^{USER}$, $\bar{\eta}^{NAV}$, $\bar{\eta}$, $\bar{\chi}_{ij}^{USER}$, $\bar{\chi}_{ij}^{NAV}$, and $\bar{\chi}_{ij}$. Then:

$$\bar{\eta} = \bar{\eta}^{USER} + \bar{\eta}^{NAV} \quad (45.38)$$

$$\bar{\chi}_{ij} = \bar{\chi}_{ij}^{USER} + \bar{\chi}_{ij}^{NAV} \quad (45.39)$$

NAVSTAR orbits are near circular, so mean values can be used for R_i , \dot{S}_i , η^{NAV} and χ_{ij}^{NAV} . When USER spacecraft orbits are near circular, mean values for r_j , \dot{s}_j , η^{USER} , and χ_{ij}^{USER} can be used. Then from Eq. 45.20:

$$\bar{\chi}_{ij} = \bar{\eta} \Delta t \quad (45.40)$$

When the USER orbit is eccentric, then a running approximation $\hat{\chi}_{ij}^{USER}$ for χ_{ij}^{USER} is required to calculate $\bar{\chi}_{ij}$:

$$\bar{\chi}_{ij} = \hat{\chi}_{ij}^{USER} + \bar{\chi}_{ij}^{NAV} \quad (45.41)$$

45.4.7 USER Orbit Partial Derivatives

From Eqs. 45.19 and 45.21:

$$\zeta_{ij} = [\rho_{ij}(t_j) - \rho_{ij}(t_j - \Delta t)] / c - \chi_{ij} \quad (45.42)$$

where ζ_{ij} has time tag-tag at t_j . Differentiate ζ_{ij} with respect to z and \dot{z} , the USER spacecraft position and velocity 3×1 component matrices:

$$\frac{\partial \zeta_{ij}}{\partial z_j} = \frac{\partial \rho_{ij}(t_j)}{\partial z_j} - \frac{\partial \rho_{ij}(t_j - \Delta t)}{\partial z_j} - \frac{\partial \chi_{ij}}{\partial z_j} \quad (45.43)$$

$$\frac{\partial \zeta_{ij}}{\partial \dot{z}_j} = \frac{\partial \rho_{ij}(t_j)}{\partial \dot{z}_j} - \frac{\partial \rho_{ij}(t_j - \Delta t)}{\partial \dot{z}_j} - \frac{\partial \chi_{ij}}{\partial \dot{z}_j} \quad (45.44)$$

where $\partial \zeta_{ij} / \partial z_j$ and $\partial \zeta_{ij} / \partial \dot{z}_j$ are 1×3 matrices, and where:

$$\frac{\partial \rho_{ij}(t_j - \Delta t)}{\partial z_j} = \frac{\partial \rho_{ij}(t_j - \Delta t)}{\partial z(t_j - \Delta t)} \frac{\partial z(t_j - \Delta t)}{\partial z(t_j)} \quad (45.45)$$

$$\frac{\partial \rho_{ij}(t_j - \Delta t)}{\partial \dot{z}_j} = \frac{\partial \rho_{ij}(t_j - \Delta t)}{\partial \dot{z}(t_j - \Delta t)} \frac{\partial \dot{z}(t_j - \Delta t)}{\partial \dot{z}(t_j)} \quad (45.46)$$

The transition matrix $\Phi(t_j - \Delta t, t_j)$ on $Z = (z, \dot{z})^T$ has the form:

$$\Phi(t_j - \Delta t, t_j) = \begin{bmatrix} \frac{\partial z(t_j - \Delta t)}{\partial z(t_j)} & \frac{\partial z(t_j - \Delta t)}{\partial \dot{z}(t_j)} \\ \frac{\partial \dot{z}(t_j - \Delta t)}{\partial z(t_j)} & \frac{\partial \dot{z}(t_j - \Delta t)}{\partial \dot{z}(t_j)} \end{bmatrix} \quad (45.47)$$

so calculate $\Phi(t_j - \Delta t, t_j)$, then extract $\partial z(t_j - \Delta t) / \partial z(t_j)$ and $\partial \dot{z}(t_j - \Delta t) / \partial \dot{z}(t_j)$ as needed. The range partials algorithm is available from the GPS range algorithm document. The χ_{ij} partials can probably be ignored due to their deweighting by c^{-2} :

$$\frac{\partial \chi_{ij}}{\partial z_j} = \int_{t_j - \Delta t}^{t_j} \frac{1}{c^2} \left\{ \mu \left[\frac{\partial r_j^{-1}}{\partial z_j} \right] + \frac{1}{2} \left[\frac{\partial \dot{s}_j^2}{\partial z_j} \right] \right\} dt$$

$$\frac{\partial \chi_{ij}}{\partial \dot{z}_j} = \int_{t_j - \Delta t}^{t_j} \frac{1}{c^2} \left\{ \mu \left[\frac{\partial r_j^{-1}}{\partial \dot{z}_j} \right] + \frac{1}{2} \left[\frac{\partial \dot{s}_j^2}{\partial \dot{z}_j} \right] \right\} dt$$

The 1×3 matrices of frequency-dependent phase count partial derivatives with respect to USER position and velocity are calculated with:

$$\frac{\partial N_j^n}{\partial z_j} = [f_n^U - \delta f_n^U] \frac{\partial \zeta_{ij}}{\partial z_j} \quad (45.48)$$

$$\frac{\partial N_j^n}{\partial \dot{z}_j} = [f_n^U - \delta f_n^U] \frac{\partial \zeta_{ij}}{\partial \dot{z}_j} \quad (45.49)$$

45.5 Ionosphere

45.5.1 Physics

Let us denote the ionospheric refractive index by n_{PH} , the group refractive index by n_{GD} , radio frequency by f , and electron density (units: number of electrons/m³) by N_e . From first principles[12], it is shown that:

$$n_{PH} = 1 - A \left(\frac{N_e}{f^2} \right) \quad (45.50)$$

$$n_{GD} = 1 + A \left(\frac{N_e}{f^2} \right) \quad (45.51)$$

where A is defined by:

$$A = \frac{e^2}{8\pi^2\epsilon_0 m_e} \quad (45.52)$$

with constants:

Symbol	Name	Constant Value
e	electron charge	$1.6021892 \times 10^{-19} C$
ϵ_0	electric permittivity	$8.854187818 \times 10^{-12} F/m$
m_e	electron mass	$10^{-31} kg$

Table 45.1: Elements of A

Notation for variables:

- N_{TEC} : total electron content
- ρ_{PH} : ionospheric phase range
- ρ_{GD} : ionospheric group delay range
- Δt_{PH} : ionospheric phase advance
- Δt_{GD} ionospheric group delay
- φ_{ION} : ionospheric phase

From Eq. 45.52:

$$A = 40.30816 \text{ (m}^3\text{Hz}^2\text{/electron)} \quad (45.53)$$

45.5.2 Total Electron Content

The total electron content N_{TEC} is the integral of electron density N_e along the path of the radio signal from transmitter to receiver. N_{TEC} values from ground to spacecraft are bounded[54]: 10^{16} electrons/m² < N_{TEC} < 10^{19} electrons/m². From Eqs. 45.50 and 45.51:

$$\rho_{PH} = -A \left(\frac{N_{TEC}}{f^2} \right) \quad (45.54)$$

$$\rho_{GD} = +A \left(\frac{N_{TEC}}{f^2} \right) \quad (45.55)$$

$$\Delta t_{PH} = \frac{\rho_{PH}}{c} = -A \left(\frac{N_{TEC}}{cf^2} \right) \quad (45.56)$$

$$\Delta t_{GD} = \frac{\rho_{GD}}{c} = +A \left(\frac{N_{TEC}}{cf^2} \right) \quad (45.57)$$

$$\varphi_{ION} = f\Delta t_{PH} = -A \left(\frac{N_{TEC}}{cf} \right) \quad (45.58)$$

$$\delta f^{ION} = d\varphi/dt = - \left[\frac{A}{cf} \right] \left[\frac{dN_{TEC}}{dt} \right] \quad (45.59)$$

Ionospheric group delay range ρ_{GD} , defined by Eq. 45.55, is inversely proportional to frequency-squared f^2 . But ionospheric phase delay frequency δf^{ION} , defined by Eq. 45.59, is inversely proportional to frequency f .

Units

Conventionally, there are two sets of units used in the literature for N_{TEC} : Natural units TECN:

$$\text{TECN} = \frac{1 \text{ electron}}{\text{m}^2} \quad (45.60)$$

in which case:

$$A = 40.30816 \left((\text{m})(\text{Hz}^2)/\text{TECN} \right) \quad (45.61)$$

and scaled units TECU:

$$\text{TECU} = \frac{1 \times 10^{16} \text{ electrons}}{\text{m}^2} \quad (45.62)$$

in which case:

$$A = 40.30816 \times 10^4 \left((\text{m})(\text{MHz}^2)/\text{TECU} \right) \quad (45.63)$$

where 1 MHz = 1×10^6 Hz. Then:

$$\Delta t_{GD} = \left(40.30816 \left((\text{m})(\text{Hz}^2)/\text{TECN} \right) \right) \left(\frac{N_{TEC}}{cf^2} \right) \quad (45.64)$$

$$\Delta t_{GD} = \left(40.30816 \times 10^4 \left((\text{m})(\text{MHz}^2)/\text{TECU} \right) \right) \left(\frac{N_{TEC}}{cf^2} \right) \quad (45.65)$$

$$\varphi_{ION} = - \left(40.30816 \left((\text{m})(\text{Hz}^2)/\text{TECN} \right) \right) \left(\frac{N_{TEC}}{cf} \right) \quad (45.66)$$

$$\varphi_{ION} = - \left(40.30816 \times 10^4 \left((\text{m})(\text{MHz}^2)/\text{TECU} \right) \right) \left(\frac{N_{TEC}}{cf} \right) \quad (45.67)$$

$$\delta f^{ION} = - \left[\frac{40.30816 \left((\text{m})(\text{Hz}^2)/\text{TECN} \right)}{cf} \right] \left[\frac{dN_{TEC}}{dt} \right] \quad (45.68)$$

$$\delta f^{ION} = - \left[\frac{40.30816 \times 10^4 \left((\text{m})(\text{MHz}^2)/\text{TECU} \right)}{cf} \right] \left[\frac{dN_{TEC}}{dt} \right] \quad (45.69)$$

45.5.3 Validation

Sections 10.8 and 10.9 of *The Handbook of Geophysics and the Space Environment*[54], by Klobuchar, presents expressions for comparison. Validate Eq. 45.64 by comparison to Klobuchar Eq. 10.70.

Insert ($c = 2.99792458 \times 10^8$ m/s) into Eq. 45.68 to get:

$$\delta f^{ION} = - \left[\frac{1.344536 \times 10^{-7}}{f} \left(\frac{(\text{Hz})(\text{cycles})}{\text{TECN}} \right) \right] \left[\frac{dN_{TEC}}{dt} \right] \quad (45.70)$$

If f is evaluated in units Hz and dN_{TEC}/dt is evaluated in units TECN/s, then δf_n^{ION} will have units Hz. Validate Eq. 45.70 by comparison to Klobuchar Eq. 10.78, noting Klobuchar's sign error.

45.6 Final Forms for Phase Count

Insert Eq. 45.59 into Eq. 45.30 to get:

$$N_j^{ni} = \bar{f}_n^N \zeta_{ij} + \bar{f}_n^N I_j^{nNi} + \bar{f}_n^U I_j^{nU} + \left[\frac{A}{c\bar{f}_n^N} \right] \int_{t_j - \Delta t}^{t_j} \left[\frac{dN_{TEC}}{dt} \right] dt + \delta N_j^{ni} \quad (45.71)$$

Define:

$$\Delta N_{TEC} = N_{TEC}(t_j) - N_{TEC}(t_j - \Delta t) \quad (45.72)$$

Then Eq. 45.71 becomes:

$$N_j^{ni} = \bar{f}_n^N \zeta_{ij} + \bar{f}_n^N I_j^{nNi} + \bar{f}_n^U I_j^{nU} + A\Delta N_{TEC} / [c\bar{f}_n^N] + \delta N_j^{ni} \quad (45.73)$$

Eq. 45.73 presents the frequency-dependent NAVSTAR-dependent Doppler phase count representation N_j^{ni} in final form, at time tag t_j . Terms on the right-hand side can be described:

- $\bar{f}_n^N \zeta_{ij}$ is the deterministic phase effect, where $\zeta_{ij} = \Delta\rho_{ij}/c - \chi_{ij}$, and χ_{ij} is the integral of the relativistic expression

- I_j^{nU} is the random frequency dependent phase phenomenology in the USER GPS receiver clock
- I_j^{nNi} is the random frequency dependent phase phenomenology in the NAVSTAR transmitter clock
- $A\Delta N_{TEC} / [c\bar{f}_n^N]$ is the random frequency dependent ionospheric effect on phase
- δN_j^{ni} is the USER receiver Costas Phase Lock Loop thermal noise, independent for each NAVSTAR link i .

45.6.1 Alternate Representation

GPS carrier phase measurements residuals, intrinsically defined in units *cycles*, are frequently expressed in the distance units of range; *e.g.*, centimeters or millimeters. There does not exist a mapping from carrier phase to range, but there does exist a mapping from carrier phase to range difference. But in so doing, one must be careful to avoid a fundamental change in semantics. Eq. 45.73 defines the phase measurement N_j^{ni} as a sum of noise-free signal $\bar{f}_n^N \zeta_{ij}$, clock noise $\bar{f}_n^N I_j^{nNi} + \bar{f}_n^U I_j^{nU}$, uncertain ionosphere perturbation $A\Delta N_{TEC} / [c\bar{f}_n^N]$, and thermal noise δN_j^{ni} . Eq. 45.75 represents signal component range difference $\Delta\rho_{ij}$ as a sum of scaled measurement $\lambda_n^N N_j^{ni}$, noise-free signal component $c\chi_{ij}$, clock noise $\bar{f}_n^N I_j^{nNi} + \bar{f}_n^U I_j^{nU}$, ionosphere perturbation $A\Delta N_{TEC} / [c\bar{f}_n^N]$, and thermal noise δN_j^{ni} . It is clear that range difference $\Delta\rho_{ij}$ is a signal component of the measurement N_j^{ni} , and in no way can be identified as a measurement – without a fundamental change in semantics.

Let λ_n denote wave length for carrier frequency \bar{f}_n^N . Then:

$$\lambda_n \bar{f}_n^N = c \quad (45.74)$$

where for L_1 and L_2 :

$$\lambda_1 = 19.029367 \frac{\text{cm}}{\text{cycle}}$$

$$\lambda_2 = 24.42102 \frac{\text{cm}}{\text{cycle}}$$

From N_j^{ni} to $\Delta\rho_{ij}$

Insert Eqs. 45.74 and 45.21 into Eq. 45.73:

$$\Delta\rho_{ij} = \lambda_n^N N_j^{ni} + c\chi_{ij} - c [I_j^{nNi} + I_j^{nU}] - \frac{A\Delta N_{TEC}}{(\bar{f}_n^N)^2} - \lambda_n^N \delta N_j^{ni} \quad (45.75)$$

where χ_{ij} , defined by Eq. 45.20, is the time integral of contributions due to special and general relativity, and where we have assumed:

$$\frac{\lambda_n^N c}{\lambda_n^U} = 1 \quad (45.76)$$

45.7 Ionosphere Removal

Consider every linear weighted combination:

$$N_j^{\text{Ei}} = aN_j^{1i} + bN_j^{2i} \quad (45.77)$$

to find expressions for undetermined unitless constants a and b , such that: (i) N_j^{Ei} is free of the ionospheric term $A\Delta N_{TEC}/[c\bar{f}_n^N]$ of Eq. 45.73, and (ii) the sum S_{ab} of measurement weights a and b is unity:

$$S_{ab} = a + b = 1 \quad (45.78)$$

Insert phase count measurement representations N_j^{1i} and N_j^{2i} of Eq. 45.73 into Eq. 45.77 to get:

$$N_j^{\text{Ei}} = (a\bar{f}_1^N + b\bar{f}_2^N) \zeta_{ij} + \left(\frac{A\Delta N_{TEC}}{c} \right) \left[\frac{a}{\bar{f}_1^N} + \frac{b}{\bar{f}_2^N} \right] + (3 \text{ other terms}) \quad (45.79)$$

Set:

$$\left[\frac{a}{\bar{f}_1^N} + \frac{b}{\bar{f}_2^N} \right] = 0 \quad (45.80)$$

to free N_j^{Ei} of the ionospheric term $A\Delta N_{TEC}/[c\bar{f}_n^N]$ in Eq. 45.79 and satisfy the first constraint on Eq. 45.77. Then Eqs. 45.80 and 45.78 yield:

$$b = - \left(\frac{\bar{f}_2^N}{\bar{f}_1^N} \right) a \quad (45.81)$$

and:

$$S_{ab} = \left(\frac{\bar{f}_1^N - \bar{f}_2^N}{\bar{f}_1^N} \right) a \quad (45.82)$$

Then from Eqs. 45.78 and 45.81, expressions for the measurement weights a and b are found:

$$a = \frac{\bar{f}_1^N}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.83)$$

$$b = - \frac{\bar{f}_2^N}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.84)$$

Then from Eq. 45.77:

$$N_j^{\text{Ei}} = \frac{\bar{f}_1^N N_j^{1i} - \bar{f}_2^N N_j^{2i}}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.85)$$

or:

$$N_j^{\text{Ei}} = \frac{\gamma N_j^{1i} - N_j^{2i}}{\gamma - 1} \quad (45.86)$$

where:

$$\gamma = \bar{f}_1^N / \bar{f}_2^N \quad (45.87)$$

Refer to N_j^{Ei} as the ionosphere-free carrier-phase measurement. Insert Eq. 45.73 into Eq. 45.86 to get the measurement representation \bar{N}_j^{Ei} to be used in the filter:

$$\bar{N}_j^{\text{Ei}} = \frac{\bar{f}_1^N H_j^{1i} - \bar{f}_2^N H_j^{2i}}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.88)$$

where ionosphere effects vanish, and where:

$$H_j^{ni} = \bar{f}_n^N \zeta_{ij} + I_j^{nNi} + I_j^{nU} + \delta N_j^{ni} \quad (45.89)$$

Define:

$$J_j^{nNi} = \bar{f}_n^N \zeta_{ij} + I_j^{nNi} + I_j^{nU} \quad (45.90)$$

so that:

$$H_j^{ni} = J_j^{nNi} + \delta N_j^{ni} \quad (45.91)$$

Then the ionosphere-free carrier-phase measurement can be written:

$$\bar{N}_j^{\text{Ei}} = S_j^{Ni} + W_j^{Ni} \quad (45.92)$$

where the measurement signal S_j^{Ni} is given by:

$$S_j^{Ni} = \frac{\bar{f}_1^N J_j^{1Ni} - \bar{f}_2^N J_j^{2Ni}}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.93)$$

and measurement white noise W_j^{Ni} is given by:

$$W_j^{Ni} = \frac{\bar{f}_1^N \delta N_j^{1i} - \bar{f}_2^N \delta N_j^{2i}}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.94)$$

Define:

$$F_1 = \frac{\bar{f}_1^N}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.95)$$

$$F_2 = \frac{\bar{f}_2^N}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.96)$$

45.7.1 Ionosphere-Free White Noise Variance for \bar{N}_j^{Ei} (Cycles)

If L_1 is used for aiding instantaneously on L_2 in the PLL, it seems possible that the serial white noise might be instantaneously cross-correlated. Let us denote the sigma on the initial measurements with σ_N , and the sigma on the ionospheric measurement with σ_W . With linear relation:

$$\sigma_W = K_{WN}\sigma_N$$

it is shown below that:

$$1.7792208 \leq K_{WN} \leq 5.7421528$$

where the left-hand value assumes maximal cross-correlation, and the right-hand value assumes zero correlation, between measurement white noise on the two frequencies.

From Eq. 45.94:

$$W_j^{Ni} = F_1\delta N_j^{1i} - F_2\delta N_j^{2i} \quad (45.97)$$

Square Eq. 45.97 and take expectation:

$$E\left\{(W_j^{Ni})^2\right\} = F_1^2 E\left\{(\delta N_j^{1i})^2\right\} + F_2^2 E\left\{(\delta N_j^{2i})^2\right\} - 2F_1F_2 E\left\{\delta N_j^{1i}\delta N_j^{2i}\right\} \quad (45.98)$$

Uncorrelated Thermal Noise with Equal Variances

If:

$$E\left\{\delta N_j^{1i}\delta N_j^{2i}\right\} = 0 \quad (45.99)$$

and:

$$E\left\{(\delta N_j^{2i})^2\right\} = E\left\{(\delta N_j^{1i})^2\right\} \quad (45.100)$$

then:

$$E\left\{(W_j^{Ni})^2\right\} = (F_1^2 + F_2^2) E\left\{(\delta N_j^{1i})^2\right\}$$

That is, the sigma σ_W on the ionospheric measurement is given by:

$$\sigma_W = \sqrt{F_1^2 + F_2^2}\sigma_N = (5.7421528)\sigma_N \quad \text{units: cycles} \quad (45.101)$$

where σ_W is defined by:

$$\sigma_W = \sqrt{E \left\{ (W_j^{Ni})^2 \right\}} \quad \text{units: cycles} \quad (45.102)$$

and where σ_N , the sigma on the initial measurements, is defined by:

$$\sigma_N = \sqrt{E \left\{ (\delta N_j^{1i})^2 \right\}} \quad \text{units: cycles} \quad (45.103)$$

Correlated Thermal Noise

Assume there exist δN_j^0 in the GPS receiver such that:

$$\delta N_j^{1i} = 154 \delta N_j^{0i} = \left(\frac{f_1}{f_0} \right) \delta N_j^{0i} \quad (45.104)$$

$$\delta N_j^{2i} = 120 \delta N_j^{0i} = \left(\frac{f_2}{f_0} \right) \delta N_j^{0i} \quad (45.105)$$

where in the NAVSTAR GPS transmitter we know:

$$f_1^N = (154) f_0^N$$

$$f_2^N = (120) f_0^N$$

Then:

$$\delta N_j^{0i} = \left(\frac{f_0}{f_1} \right) \delta N_j^{1i} = \left(\frac{f_0}{f_2} \right) \delta N_j^{2i}$$

and:

$$E \left\{ (W_j^{Ni})^2 \right\} = \frac{(f_1 + f_2)^2}{f_0^2} E \left\{ (\delta N_j^{0i})^2 \right\} \quad (45.106)$$

or:

$$\sigma_W = \left(\frac{f_1 + f_2}{f_0} \right) \sigma_{N0} \quad \text{units: cycles} \quad (45.107)$$

$$\frac{f_1 + f_2}{f_0} = 274$$

where:

$$\sigma_W = \sqrt{E \left\{ (W_j^{Ni})^2 \right\}} \quad \text{units: cycles}$$

$$\sigma_{N0} = \sqrt{E \left\{ (\delta N_j^{0i})^2 \right\}} \quad \text{units: cycles}$$

Since:

$$\delta N_j^{0i} = \left(\frac{f_0}{f_1}\right) \delta N_j^{1i} = \left(\frac{f_0}{f_2}\right) \delta N_j^{2i}$$

then:

$$\sqrt{E \{(\delta N_j^{0i})^2\}} = \left(\frac{1}{154}\right) \sqrt{E \{(\delta N_j^{1i})^2\}} = \left(\frac{1}{120}\right) \sqrt{E \{(\delta N_j^{2i})^2\}}$$

Therefore:

$$\sigma_W = \left(\frac{\sigma_{N1}}{154}\right) = \left(\frac{\sigma_{N2}}{120}\right) \quad \text{units: cycles}$$

where:

$$\sigma_{N1} = \sqrt{E \{(\delta N_j^{1i})^2\}} \quad \text{units: cycles}$$

$$\sigma_{N2} = \sqrt{E \{(\delta N_j^{2i})^2\}} \quad \text{units: cycles}$$

45.7.2 Ionosphere-Free White Noise Variance for $\Delta\rho$ (cm)

Eq. 45.85 can be written:

$$N_j^{\text{Ei}} = \frac{\lambda_2 N_j^{1i} - \lambda_1 N_j^{2i}}{\lambda_2 - \lambda_1} \quad (45.108)$$

where $\lambda_n f_n = c$. Insert Eq. 45.75 into Eq. 45.108 to get:

$$\Delta\rho_{ij} = \Lambda_S N_j^{\text{Ei}} + c\chi_{ij} - c\Lambda_1 [I_j^{1Ni} + I_j^{1U}] + c\Lambda_2 [I_j^{2Ni} + I_j^{2U}] - \Lambda_{W1} \delta N_j^{1i} + \Lambda_{W2} \delta N_j^{2i} \quad (45.109)$$

where the wave-length coefficients are defined and evaluated:

$$\Lambda_{W1} = \frac{\lambda_1 \lambda_2^2}{\lambda_2^2 - \lambda_1^2} = 48.443594 \text{ cm/cycle; thermal noise } L_1 \quad (45.110)$$

$$\Lambda_{W2} = \frac{\lambda_1^2 \lambda_2}{\lambda_2^2 - \lambda_1^2} = 37.748257 \text{ cm/cycle; thermal noise } L_2 \quad (45.111)$$

$$\Lambda_S = \frac{\lambda_1 \lambda_2}{\lambda_2 + \lambda_1} = 10.695338 \text{ cm/cycle; signal} \quad (45.112)$$

$$\Lambda_1 = \frac{\lambda_2^2}{\lambda_2^2 - \lambda_1^2} = 2.5457281 \text{ (unitless); clocks } L_1 \quad (45.113)$$

$$\Lambda_2 = \frac{\lambda_1^2}{\lambda_2^2 - \lambda_1^2} = 1.5457281 \text{ (unitless); clocks } L_2 \quad (45.114)$$

and where ionospheric terms subtract out¹². Note that N_j^{Ei} and χ_{ij} are deterministic signals. Thermal noises δN_j^{2i} and δN_j^{1i} are independent of clock noises I_j^{nNi} and I_j^{nU} (derived from oscillators) because thermal noise is white noise. The sizes of white noise coefficients $\Lambda_{W1} = 48.443594$ cm/cycle and $\Lambda_{W2} = 37.748257$ cm/cycle, relative to signal $\Lambda_S = 10.695338$ cm/cycle, are somewhat alarming.

Define:

$$W = \Lambda_{W2} \delta N_j^{2i} - \Lambda_{W1} \delta N_j^{1i} \quad (45.115)$$

Then:

$$\begin{aligned} E \{W^2\} &= \left(\frac{\lambda_1 \lambda_2^2}{\lambda_2^2 - \lambda_1^2} \right)^2 E \{(\delta N_j^{1i})^2\} + \left(\frac{\lambda_1^2 \lambda_2}{\lambda_2^2 - \lambda_1^2} \right)^2 E \{(\delta N_j^{2i})^2\} \\ &\quad - 2 \left(\frac{\lambda_1 \lambda_2^2}{\lambda_2^2 - \lambda_1^2} \right) \left(\frac{\lambda_1^2 \lambda_2}{\lambda_2^2 - \lambda_1^2} \right) E \{\delta N_j^{1i} \delta N_j^{2i}\} \end{aligned} \quad (45.116)$$

Uncorrelated Thermal Noise with Equal Variances

If:

$$E \{\delta N_j^{1i} \delta N_j^{2i}\} = 0$$

and:

$$E \{(\delta N_j^{2i})^2\} = E \{(\delta N_j^{1i})^2\}$$

then:

$$E \{W^2\} = [\Lambda_{W1}^2 + \Lambda_{W2}^2] E \{(\delta N_j^{1i})^2\}$$

and:

$$\sigma_W = \sqrt{\Lambda_{W1}^2 + \Lambda_{W2}^2} \sigma_N$$

$$\sqrt{\Lambda_{W1}^2 + \Lambda_{W2}^2} = 61.414271 \text{ cm/cycle}$$

where:

¹²The equation above was derived in mid 2004, and then rederived on 21 Mar 05 – with same result.

$$\sigma_W = \sqrt{E \{W^2\}}$$

$$\sigma_N = \sqrt{E \{(\delta N_j^{1i})^2\}}$$

Correlated Thermal Noise

$$\begin{aligned} E \{W^2\} &= \left(\frac{\lambda_1 \lambda_2^2}{\lambda_2^2 - \lambda_1^2} \right)^2 E \{(\delta N_j^{1i})^2\} + \left(\frac{\lambda_1^2 \lambda_2}{\lambda_2^2 - \lambda_1^2} \right)^2 E \{(\delta N_j^{2i})^2\} \\ &\quad - 2 \left(\frac{\lambda_1 \lambda_2^2}{\lambda_2^2 - \lambda_1^2} \right) \left(\frac{\lambda_1^2 \lambda_2}{\lambda_2^2 - \lambda_1^2} \right) E \{\delta N_j^{1i} \delta N_j^{2i}\} \end{aligned}$$

Now δN_j^{1i} and δN_j^{2i} denote thermal noise, in units cycles, on L_1 and L_2 , and W denotes thermal noise after ionosphere combination in units cm.

$$W = \Lambda_{W2} \delta N_j^{2i} - \Lambda_{W1} \delta N_j^{1i}$$

$$= \Delta \rho_{ij} - \{\Lambda_S N_j^{\text{Ei}} + c \chi_{ij} - c \Lambda_1 [I_j^{1Ni} + I_j^{1U}] + c \Lambda_2 [I_j^{2Ni} + I_j^{2U}]\}$$

$$\delta N_j^{1i} = 154 \delta N_j^0$$

$$\delta N_j^{2i} = 120 \delta N_j^0$$

$$E \{W^2\} = \lambda_0^2 E \{(\delta N_j^0)^2\}$$

Thus:

$$\sigma_W = \lambda_0 \sigma_{N0}$$

where:

$$\lambda_0 = 2930.522 \text{ cm/cycle}$$

$$\sigma_W = \sqrt{E \{W^2\}}$$

$$\sigma_{N0} = \sqrt{E \{(\delta N_j^0)^2\}}$$

Now:

$$\lambda_0 = 154 \lambda_1 = 120 \lambda_2$$

and:

$$\delta N_j^{1i} = 154 \delta N_j^0$$

$$\delta N_j^{2i} = 120 \delta N_j^0$$

So:

$$E \{ (\delta N_j^0)^2 \} = \left(\frac{1}{154} \right)^2 E \{ (\delta N_j^{1i})^2 \} = \left(\frac{1}{120} \right)^2 E \{ (\delta N_j^{2i})^2 \}$$

and:

$$\sqrt{E \{ (\delta N_j^0)^2 \}} = \left(\frac{1}{154} \right) \sqrt{E \{ (\delta N_j^{1i})^2 \}} = \left(\frac{1}{120} \right) \sqrt{E \{ (\delta N_j^{2i})^2 \}}$$

$$\sigma_{N0} = \left(\frac{1}{154} \right) \sigma_{N1} = \left(\frac{1}{120} \right) \sigma_{N2}$$

Therefore:

$$\sigma_W = \lambda_1 \sigma_{N1} = \lambda_2 \sigma_{N2} \quad (45.117)$$

$$\sigma_W = (19.029367 \text{ cm/cycle}) \sigma_{N1} = (24.42102 \text{ cm/cycle}) \sigma_{N2}$$

where:

$$\sigma_{N1} = \sqrt{E \{ (\delta N_j^{1i})^2 \}}$$

$$\sigma_{N2} = \sqrt{E \{ (\delta N_j^{2i})^2 \}}$$

45.8 First Differences on N_j^{Ei}

45.8.1 Remove USER Clock Phase Perturbations

Define:

$$\Delta N_j^E = N_j^{Ep} - N_j^{Eq} \quad (45.118)$$

where p and q refer to distinct NAVSTARS. Insert Eq. 45.88 into Eq. 45.118 to get:

$$(\bar{f}_1^N - \bar{f}_2^N) \Delta N_j^E = [\bar{f}_1^N H_j^{1p} - \bar{f}_2^N H_j^{2p}] - [\bar{f}_1^N H_j^{1q} - \bar{f}_2^N H_j^{2q}] \quad (45.119)$$

Insert Eq. 45.89 into Eq. 45.119 to get:

$$\Delta N_j^E = R_j^E + S_j^E \quad (45.120)$$

where:

$$R_j^E = \frac{\bar{f}_1^N [I_j^{1Np} - I_j^{1Nq}] - \bar{f}_2^N [I_j^{2Np} - I_j^{2Nq}] + \bar{f}_1^N [\delta N_j^{1p} - \delta N_j^{1q}] - \bar{f}_2^N [\delta N_j^{2p} - \delta N_j^{2q}]}{\bar{f}_1^N - \bar{f}_2^N} \quad (45.121)$$

$$S_j^E = (\bar{f}_1^N + \bar{f}_2^N) [\zeta_{pj} - \zeta_{qj}] \quad (45.122)$$

and where $\bar{f}_1^N I_j^{1U}$ and $\bar{f}_2^N I_j^{2U}$ are differenced out. That is, USER clock phase perturbations I_j^{1U} and I_j^{2U} vanish entirely due to first differences ΔN_j^E on ionosphere-free Doppler phase count measurement representations. NAVSTAR clock phase perturbations I_j^{1Np} , I_j^{2Np} , I_j^{1Nq} , and I_j^{2Nq} all survive, and all USER receiver PLL thermal noise perturbations δN_j^{ni} survive.

45.8.2 Root-Variance Mappings

Since USER receiver PLL thermal white noise is independent for each NAVSTAR, all root-variances for singly differenced ionosphere measurements are obtained by multiplication of the root-variances for ionosphere measurements by $\sqrt{2}$. Let σ_Δ denote white noise root-variance of differenced ionosphere measurements. Thus:

$$\sigma_\Delta = \sqrt{2}\lambda_1\sigma_{N1} = \sqrt{2}\lambda_2\sigma_{N2} \quad (45.123)$$

$$\sigma_\Delta = (26.911589 \text{ cm/cycle}) \sigma_{N1} = (34.536538 \text{ cm/cycle}) \sigma_{N2}$$

where:

$$\sigma_{N1} = \sqrt{E \{ (\delta N_j^{1i})^2 \}}$$

$$\sigma_{N2} = \sqrt{E \{ (\delta N_j^{2i})^2 \}}$$

$$\sigma_\Delta = \sqrt{2E \{ W^2 \}}$$

45.9 Example

Suppose:

$$\sigma_{\Delta} = 1 \text{ cm}$$

Then:

$$\sigma_{N1} = \frac{1 \text{ cm}}{26.911589 \text{ cm/cycle}} = 0.0371587 \text{ cycles}$$

$$\sigma_{N2} = \frac{1 \text{ cm}}{34.536538 \text{ cm/cycle}} = 0.0289548 \text{ cycles}$$

45.10 Partials and Covariance

In symmetry with Eqs. 45.5 through 45.10, define:

$$\bar{f}_0^N = (10.23 \text{ MHz}) (1 - 4.4647 \times 10^{-10}) \quad (45.124)$$

USER clock frequencies can be sufficiently approximated with:

$$f_1^U = \bar{f}_1^U + \delta f_1^U = (154) f_0^U \quad (45.125)$$

$$f_2^U = \bar{f}_2^U + \delta f_2^U = (120) f_0^U \quad (45.126)$$

where:

$$f_0^U = \bar{f}_0^U + \delta f_0^U \quad (45.127)$$

$$\delta f_1^U = (154) \delta f_0^U \quad (45.128)$$

$$\delta f_2^U = (120) \delta f_0^U \quad (45.129)$$

where f_0^U is generated by the USER spacecraft oscillator, and where δf_0^U is a random perturbation on the nominal frequency \bar{f}_0^U .

45.10.1 Single Frequency

Recall Eq. 45.73:

$$N_j^{ni} = (\bar{f}_n^N + \delta f_n^N) \zeta_{ij} + \delta f_n^U \Delta t + \delta N_j^{ni} + A \Delta N_{TEC} / [c f_n^N]$$

Partials

$$\frac{\partial N_j^{ni}}{\partial \delta f_n^U} = \Delta t \quad (45.130)$$

$$\frac{\partial N_j^{ni}}{\partial \delta f_n^N} = \zeta_{ij} \quad (45.131)$$

Covariance

$$P_{N_j^{ni}} = E \left\{ (N_j^{ni})^2 \right\}$$

$$P_{N_j^{ni}} = (\zeta_{ij})^2 E \left\{ (\delta f_n^N)^2 \right\} + (\Delta t)^2 E \left\{ (\delta f_n^U)^2 \right\} + E \left\{ (\delta N_j^{ni})^2 \right\} \quad (45.132)$$

45.10.2 Two-Frequency Ionosphere Removal

Insert Eqs. 45.125 and 45.126 into Eq. 45.88, and then use Eq. 45.89 and Eqs. 45.128 and 45.129 to get:

$$N_j^{\text{Ei}} =$$

$$\left(\frac{1}{34} \right) \left[\left((154)^2 - (120)^2 \right) \left[\Delta t \delta f_0^U + (\bar{f}_0^N + \delta f_0^N) \zeta_{ij} \right] + \left[(154) \delta N_j^{1i} - (120) \delta N_j^{2i} \right] \right] \quad (45.133)$$

Partials

Define unitless constant B :

$$B = \frac{\left[(154)^2 - (120)^2 \right]}{(34)} = 274 \quad (45.134)$$

Estimate USER clock frequency perturbation δf_0^U :

$$\frac{\partial N_j^{\text{Ei}}}{\partial \delta f_0^U} = B \Delta t \quad (45.135)$$

Estimate NAVSTAR clock frequency perturbation δf_0^N :

$$\frac{\partial N_j^{\text{Ei}}}{\partial \delta f_0^N} = B \zeta_{ij} \quad (45.136)$$

Covariance

Define:

$$P_{\text{Ei } j} = E \left\{ (N_j^{\text{Ei}})^2 \right\} \quad (45.137)$$

Insert Eq. 45.133 into Eq. 45.137. Then:

$$\begin{aligned} P_{\text{Ei } j} &= B^2 (\Delta t)^2 E \left\{ (\delta f_0^U)^2 \right\} + B^2 \zeta_{ij}^2 E \left\{ (\delta f_0^N)^2 \right\} \\ &+ \left(\frac{154}{34} \right)^2 E \left\{ (\delta N_j^{1i})^2 \right\} + \left(\frac{120}{34} \right)^2 E \left\{ (\delta N_j^{2i})^2 \right\} \end{aligned} \quad (45.138)$$

45.10.3 First Differences on N_j^{Ei}

$$\Delta N_j^E = (\bar{f}_1^N + \bar{f}_2^N) [\zeta_{pj} - \zeta_{qj}] + \Delta \delta_j^E \quad (45.139)$$

$$\Delta \delta_j^E = (B [\zeta_{pj} - \zeta_{qj}] (\delta f_0^N) + \Delta W_j^E) \quad (45.140)$$

$$\Delta W_j^E = \left(\frac{154}{34} \right) (\delta N_j^{1p} - \delta N_j^{1q}) - \left(\frac{120}{34} \right) (\delta N_j^{2p} - \delta N_j^{2q}) \quad (45.141)$$

Partials

$$\frac{\partial \Delta N_j^E}{\partial \delta f_0^N} = B [\zeta_{pj} - \zeta_{qj}] \quad (45.142)$$

Covariance

$$E \left\{ (\Delta N_j^E)^2 \right\} = E \left\{ (\Delta \delta_j^E)^2 \right\} \quad (45.143)$$

$$E \left\{ (\Delta \delta_j^E)^2 \right\} = B^2 [\zeta_{pj} - \zeta_{qj}]^2 E \left\{ (\delta f_0^N)^2 \right\} + E \left\{ (\Delta W_j^E)^2 \right\} \quad (45.144)$$

$$\begin{aligned} E \left\{ (\Delta W_j^E)^2 \right\} &= \left(\frac{154}{34} \right)^2 \left[E \left\{ (\delta N_j^{1p})^2 \right\} + E \left\{ (\delta N_j^{1q})^2 \right\} \right] \\ &+ \left(\frac{120}{34} \right)^2 \left[E \left\{ (\delta N_j^{2p})^2 \right\} + E \left\{ (\delta N_j^{2q})^2 \right\} \right] \end{aligned} \quad (45.145)$$

45.11 Receiver Clock Error Model

We shall assume that the receiver clock is a state-of-the-art oven-controlled crystal clock.

AGI resources have been applied to simulate and graph appropriate ensembles of clock modeling stochastic sequences in clock phase x_j and fractional frequency \bar{y}_j , and overlay the graphs with associated two-sigma boundary curves. Mutual validation of simulations and covariance calculations has been achieved with a demonstration of consistency between an ensemble of simulations and its two-sigma boundaries.

45.11.1 Covariance on Clock Phase

Our physical clock model derives from Allan[1], Table II, page 651. Values for clock dependent parameter constants a_0 and a_{-2} define selected models for composition of the Allan variance $\sigma_y^2(\tau)$ according to Eqs. 45.151 and 45.154. We assume that the receiver clock can be sufficiently characterized¹³ by frequency modulated (FM) white noise and frequency modulated random walk. Then clock phase modulation (PM) will be driven by their generalized integrals.

Let $x(t)$ denote clock time deviation (or clock phase deviation) and let $y(t)$ denote clock frequency deviation. Formally, Allan[1] presents the (generalized¹⁴) integral:

$$x(t) = \int_0^t y(\eta) d\eta \quad (45.146)$$

Let τ denote clock sample time. The sample Allan variance is defined by:

$$\sigma_y^2(\tau) = \lim_{n \rightarrow \infty} \frac{1}{2n} \sum_{j=1}^n (\bar{y}_{j+1} - \bar{y}_j)^2 \quad (45.147)$$

where:

$$\bar{y}_j = \frac{x_{j+1} - x_j}{\tau} \quad (45.148)$$

$$\tau = t_{j+1} - t_j \quad (45.149)$$

$$x_j = x(t_j) \quad (45.150)$$

¹³This assumption is supported by Allan: "Note that the quadratic D term occurs because $x(t)$ is the integral of $y(t)$, the fractional frequency, and is often the predominant cause of time deviation."

¹⁴Allan does not distinguish between integrals and generalized integrals in his paper referenced above. But the standard calculus does not work for the "integration" of white noise and random walk sequences. These sequences are summable and differenceable, but they are not integrable and differentiable. It is nonetheless helpful to communicate the relation between x and y by analogy to the standard calculus in the absence of measure theory and Lebesgue integration tools.

Frequency White Noise

The physical Allan variance for FM white noise is given by:

$$\sigma_y^2(\tau) = a_0\tau^{-1}, \text{ FM white noise} \quad (45.151)$$

Compare Eqs. 45.147 and 45.151 to write:

$$a_0\tau^{-1} = \lim_{n \rightarrow \infty} \frac{1}{2n} \sum_{j=1}^n (\bar{y}_{j+1} - \bar{y}_j)^2 \quad (45.152)$$

From this equation derive:

$$E\{x(t_k)x(t_j)\} = (a_0)(\min[t_j, t_k]) \quad (45.153)$$

A candidate value: $a_0 = 10^{-23} s$.

Frequency Random Walk

The physical Allan variance for FM random walk is given by:

$$\sigma_y^2(\tau) = a_{-2}\tau, \text{ FM random walk} \quad (45.154)$$

Compare Eqs. 45.147 and 45.154 to write:

$$a_{-2}\tau = \lim_{n \rightarrow \infty} \frac{1}{2n} \sum_{j=1}^n (\bar{y}_{j+1} - \bar{y}_j)^2 \quad (45.155)$$

Adopt the reset to zero condition:

$$y_j = 0, t_j = 0 \quad (45.156)$$

Then derive:

$$E\{(y(t))^2\} = 3a_{-2}t \quad (45.157)$$

A candidate value: $a_{-2} = (10^{-25}/500) s^{-1}$.

Superposition

Ignoring FM flicker noise and phase modulated (PM) sequences given by Allan, then:

$$\sigma_y^2(\tau) = a_0\tau^{-1} + a_{-2}\tau \quad (45.158)$$

45.11.2 Simulations

FM White Noise

Let $t_0 = 0$, τ_0 a positive small time increment, x_0 a positive small number, and n an unbounded positive integer. Define:

$$t = n\tau_0 > t_0 \quad (45.159)$$

$$x(t) = \sum_{i=1}^n x_i \quad (45.160)$$

and adopt the probability statement:

$$P\{x_i = x_0\} = P\{x_i = -x_0\} = 0.5 \quad (45.161)$$

Then [93]:

$$E\{(x(t))^2\} = \left(\frac{x_0^2}{\tau_0}\right)t \quad (45.162)$$

when the sizes of τ_0 and x_0 are sufficiently small. From Eq. 45.153:

$$E\{(x(t))^2\} = a_0 t \quad (45.163)$$

Comparison of the latter two equations yields:

$$x_0 = \sqrt{a_0 \tau_0} \quad (45.164)$$

Given $t_0 = 0$, and values for a_0 and τ_0 , derive x_0 from Eq. 45.164, and with the help of a uniform random number generator, simulate the random walk sequence in clock phase, defined by Eq. 45.160, due to FM white noise. Derive the corresponding white noise sequence in clock frequency with the help of Eq. 45.148.

FM Random Walk

Let $t_0 = 0$, τ_0 a positive small time increment, y_0 a positive small number, and n an unbounded positive integer. Define:

$$t = n\tau_0 > t_0 \quad (45.165)$$

$$y(t) = \sum_{i=1}^n y_i \quad (45.166)$$

and adopt the probability statement:

$$P\{y_i = y_0\} = P\{y_i = -y_0\} = 0.5 \quad (45.167)$$

Then [93]:

$$E \left\{ (y(t))^2 \right\} = \left(\frac{y_0^2}{\tau_0} \right) t \quad (45.168)$$

when the sizes of τ_0 and y_0 are sufficiently small. From Eq. 45.157:

$$E \left\{ (y(t))^2 \right\} = 3a_{-2}t \quad (45.169)$$

Comparison of the latter two equations yields:

$$y_0 = \sqrt{3a_{-2}\tau_0} \quad (45.170)$$

Given $t_0 = 0$, and values for a_{-2} and τ_0 , derive y_0 from Eq. 45.170, and with the help of a uniform random number generator, simulate the random walk sequence in clock frequency $y(t)$, defined by Eq. 45.166, due to FM random walk.

Clock phase, due to FM random walk is derived by inserting Eq. 45.166 into Eq. 45.146:

$$x(t) = y(t)\tau_0 \quad (45.171)$$

45.11.3 Filter Time Update for Receiver Clock

Let t_k and $t_{k+1} > t_k$ define the time update interval $[t_k, t_{k+1}]$. Let us denote the receiver clock filter transition function $\Phi_{k+1,k}$, and the receiver clock filter process noise covariance $Q_{k+1,k}$.

State Estimate

State estimate propagation for the receiver clock:

$$X_{k+1|k} = \Phi_{k+1,k}X_{k|k} \quad (45.172)$$

where:

$$\Phi_{k+1,k} = \begin{bmatrix} 1 & (t_{k+1} - t_k) \\ 0 & 1 \end{bmatrix} \quad (45.173)$$

Error Covariance

Error covariance propagation for the receiver clock:

$$P_{k+1|k} = \Phi_{k+1,k}P_{k|k}\Phi_{k+1,k}^T + Q_{k+1,k} \quad (45.174)$$

where:

$$Q_{k+1,k} =$$

$$\begin{bmatrix} a_0(t_{k+1} - t_k) + a_{-2}(t_{k+1} - t_k)^3 & 3a_{-2}(t_{k+1} - t_k)^2/2 \\ 3a_{-2}(t_{k+1} - t_k)^2/2 & 3a_{-2}(t_{k+1} - t_k) \end{bmatrix} \quad (45.175)$$

$$P_{j|k} = \begin{bmatrix} P_{j|k}^x & P_{j|k}^{xy} \\ P_{j|k}^{yx} & P_{j|k}^y \end{bmatrix}, \quad j \in \{k, k+1\} \quad (45.176)$$

45.11.4 State Estimate Parameters

We shall estimate the fractional frequency $y(t)$ and its generalized integral $x(t)$ for the user spacecraft clock.

Chapter 46

GPS Navigation Solution

46.1 Introduction

ODTK can convert GPS pseudorange observations into so-called "navigation solutions." A navigation solution is the estimated position and clock offset of a GPS receiver based on four (4) or more pseudoranges at a single epoch. The navigation solutions of the USER spacecraft receiver can be used as an alternative measurement type and are akin to "normal places" for GPS measurements. Advantages of navigation solutions are that they are faster and simpler to process for orbit determination purposes; disadvantages are that some information may be inherently lost as a result of the combination and their observational uncertainty may be harder to characterize.

Bancroft [5] introduced an analytic method of navigation solution that provides position and clock offset given four (4) pseudoranges at the same time. Yang and Chen [133] provided an extension to Bancroft's method so that more than four pseudoranges may be used. This latter algorithm has been adopted for use in ODTK.

46.2 Algorithm

Let $n \geq 4$ be the number of GPS space-vehicle transmitters observed at the time of the navigation solution. Let B be a matrix having four columns and n rows. Let the first three elements of each row of B contain the position of the i^{th} GPS space vehicle, and let the fourth element contain the corrected pseudorange ρ_i (c.f. Eq. (44.1)) corresponding with the i^{th} space vehicle, *e.g.*,

$$B = \begin{bmatrix} x_1 & y_1 & z_1 & \rho_1 \\ x_2 & y_2 & z_2 & \rho_2 \\ x_3 & y_3 & z_3 & \rho_3 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & \rho_n \end{bmatrix}. \quad (46.1)$$

Define the M matrix as

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \quad (46.2)$$

and generally define the Lorentz inner product for two 4-space vectors a and b as

$$\langle a, b \rangle = a^T M b = a_1 b_1 + a_2 b_2 + a_3 b_3 - a_4 b_4. \quad (46.3)$$

Let the USER location and ranging bias of the receiver be represented by the column vector $r = [x \ y \ z \ c\Delta t]^T$, where c is the speed of light and Δt is the receiver clock offset relative to GPS system time. The navigation solution is therefore computed as

$$r = M B^{-1} (\Lambda \tau + \alpha), \quad (46.4)$$

where $\tau = [1 \ 1 \ 1 \ \cdots \ 1]^T$ is a column vector of length n . Each element of the vector α is computed from the Lorentz inner product (Eq. (46.3)) of each of the 4-space row vectors in B (Eq. 46.1) with itself, such that

$$\alpha = \frac{1}{2} \begin{bmatrix} x_1 x_1 + y_1 y_1 + z_1 z_1 - \rho_1 \rho_1 \\ x_2 x_2 + y_2 y_2 + z_2 z_2 - \rho_2 \rho_2 \\ x_3 x_3 + y_3 y_3 + z_3 z_3 - \rho_3 \rho_3 \\ \vdots \\ x_n x_n + y_n y_n + z_n z_n - \rho_n \rho_n \end{bmatrix}. \quad (46.5)$$

Finally, the scalar Λ in Eq. 46.4 is a root of the quadratic equation

$$0 = E\Lambda^2 + 2F\Lambda + G, \quad (46.6)$$

where

$$E = \left\langle (B^T B)^{-1} B^T \tau, (B^T B)^{-1} B^T \tau \right\rangle, \quad (46.7)$$

$$F = \left\langle (B^T B)^{-1} B^T \tau, (B^T B)^{-1} B^T \alpha \right\rangle - 1, \quad (46.8)$$

$$G = \left\langle (B^T B)^{-1} B^T \alpha, (B^T B)^{-1} B^T \alpha \right\rangle \quad (46.9)$$

The solution of Λ is therefore

$$\Lambda = \frac{-F \pm \sqrt{F^2 - 4EG}}{2E}, \quad (46.10)$$

However, only one of the two roots results in a correct location. Presently, ODTK chooses the root that minimizes the magnitude of the computed clock offset, *i.e.*, $|\Delta t|$.

A detailed explanation and justification of the methodology can be found in [5] and [133].

Part XII

Global Positioning System (GPS) Ground Receivers

Chapter 47

GPS Composite Clock

47.1 GOMA-E

The GPS Composite Clock Model has been implemented in ODTK according to GOMA-E[110], Section 10.3.2.3.5.10. The GPS clock steering algorithm has been implemented in ODTK according to GOMA-E, Section 10.3.2.3.5.11. Distribution of GOMA-E is authorized for US DOD and its contractors only.

The Zucca-Tavella[105] three-state clock model was used in place of the two-state model referenced in GOMA-E.

47.2 Tutorial

The GPS Composite Clock model derives from a paper by Kenneth Brown[11]. This paper is very difficult to read. Therefore I have prepared a tutorial, derived from simulations, to aid in understanding the GPS Composite Clock.

47.2.1 Simulation Narrative

Simulate Random Clock Deviations

First three random clock parameter deviation functionals were simulated for each of four clocks across an eight-day interval using the Zucca-Tavella[105] clock model. This simulation embedded the evaluation of Wiener processes and the evaluation of stochastic integrals on Wiener process differentials according to Zucca-Tavella.

See Fig. 47.1 for an overlay of simulations of clock phase deviations for the four clocks. Clocks S1 and S2 are GPS ground station clocks, and clocks N1 and N2 are spaceborne GPS NAVSTAR clocks.

Simulate GPS Range Measurements

Then GPS range¹ measurements were simulated using the simulated random clock phase² deviation functionals.

Estimate Random Clock Deviations with KF1

I estimated the three random clock parameter deviation functionals from simulated GPS range measurements for each of the four clocks using a Kalman filter, call it KF1, with a 12×1 matrix clock state estimate structure. Also, I simulated filter initial condition errors. See Fig. 47.2 for an overlay of simulated and estimated phase deviation functionals for the four clocks. Visual inspection of Fig. 47.2 reveals simulated and estimated phase deviation patterns with definite symmetries.

The clock parameter deviation functionals are not observable from the GPS range measurements – after filter initialization. That is, the error variance on each clock parameter deviation is never reduced due to processing a GPS range measurement – after filter initialization. See Fig. 47.5 for a demonstration of non-observability and unboundedness of clock phase deviation for clock S1, and notice that the phase deviation estimate is consistent with its error covariance. Related graphs for the other three clocks present the same behavior.

Form Clock Variations

I formed difference time functionals by subtracting the estimated clock parameter deviation functionals from the simulated clock parameter deviation functionals. I refer to these differences as clock parameter *variations*.

47.2.2 Similarity in Clock Parameter Variations

A remarkable phenomenon is observed: The clock parameter variations of each clock are similar to those of all other clocks in:

- phase
- frequency
- frequency drift

See Figs. 47.3 and 47.4 for a demonstration of clock phase variation similarity with filter initial condition errors simulated. Identify each of the four clock parameter variations as an *estimate* of the **U**nobserved **V**ariation **C**ommon to all **C**locks – the UVCC³. The Kalman filter KF1, applied to simulated GPS range measurements, has enabled the calculation of four estimates of the UVCC.

¹I use *GPS range* for *GPS pseudo-range* for clarity in the sequel.

²Phase deviation is the first of the three random clock parameter deviations.

³Ken Brown[11] refers to the UVCC as the Implicit Ensemble Mean (IEM).

Absence of Filter Initial Condition Errors

Fig. 47.8 presents the four similar estimates of the UVCC in the absence of simulated filter initial condition errors. Compare Fig. 47.8 to Fig. 47.3 to see the effect of filter initial condition errors on similar estimates of the UVCC.

47.2.3 Optimal Estimation of the UVCC

Now at each time-tag of each GPS range measurement, sequentially calculate the mean of the ensemble of UVCC estimates, given the clock variation values in hand. In other words calculate the conditional mean over the four clock ensemble treating each 3×1 matrix clock variation value at each GPS range time-tag as a UVCC pseudo-measurement. Invoke Sherman's Theorem to employ the Kalman filter a second time to sequentially generate optimal estimates of the UVCC time function from newly-identified UVCC pseudo-measurements. Refer to this Kalman filter as KF2.

Here the Kalman filter is functionally equivalent to the method of sequential least squares because there is no random forcing function driving its state estimate. One could invoke sequential least squares or batch least squares to estimate the UVCC. The latter was Ken Brown's approach[11]. See Figs. 47.6 and 47.7 for a demonstration of Brown's Transparent Variations.

My proposed improvement is to use the Kalman filter KF2 rather than batch least squares. Here the UVCC pseudo-measurement size is 3×1 rather than $3N \times 1$ for N clocks, and thus provides a very significant computational advantage for $N \geq 24$ over Ken Brown's approach with *Transparent Variations*. Further, there is no arbitrary add-hoc degree of freedom as in the GOMA-E pseudo-measurement approach. For my proposed improvement, the UVCC conditional mean is appropriately calculated at each time-tag of each GPS range measurement.

Contrast the Two Kalman Filters

KF1 calculates optimal sequential unobservable estimates of the three random GPS clock parameter deviation functionals from GPS range measurements. KF2 calculates optimal sequential observable estimates of the UVCC from UVCC pseudo-measurements.

47.2.4 Real World Application

Clock parameter variations are differences between unknown true clock parameter values and known estimated clock parameter values. Clock parameter variations are unknown in the real world because true clock parameter values are unknown. Thus the UVCC is an unknown clock parameter variation.

The $3N \times 3N$ matrix (12×12 here) clock parameter state estimate error covariance function⁴ $P_{k|k}$ for KF1 is known in the real world. The 3×3 matrix

⁴ $P_{n|m}$ denotes the covariance function at epoch t_n with last measurement processed at

pseudo-measurement residual covariance function, denoted here by $C_{k|k}^{UVCC}$, for each KF2 UVCC pseudo-measurement is given by a 3×3 sub-matrix from $P_{k|k}$. This is all that is required to calculate the 3×3 state estimate error covariance $P_{k|k}^{UVCC}$ for the UVCC estimate. Thus $P_{k|k}^{UVCC}$ is also known in the real world. The evaluation of $P_{k|k}^{UVCC}$ statistically quantifies the difference between each KF1 estimate of UVCC and the unknown UVCC.

Unbounded eigenvalues of $P_{k|k}$ would eventually destroy KF1. Simulated UVCC pseudo-measurements, observable to the KF1 state, could be processed with GPS range measurements by KF1 so as to produce an appropriately bounded covariance matrix $\tilde{P}_{k|k}$ function for use in KF1. But it is sufficient to perform the measurement update on the KF1 state estimate error covariance $P_{k|k}$ using the pseudo-measurement residual covariance function $C_{k|k}^{UVCC}$, without actually simulating a UVCC measurement. Recursively applied, this would generate a bounded $\tilde{P}_{k|k}$ for real world application.



Figure 47.1: Sim Phase Dev (units: 1×10^{-7} s) vs Time (0 - 8) d

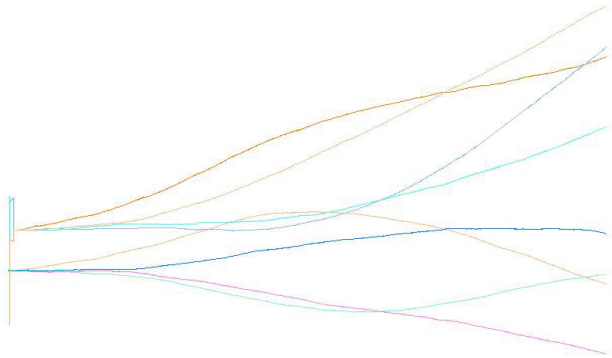


Figure 47.2: Sim & Est Phase Dev (units: 1×10^{-7} s) vs Time (0 - 8) d

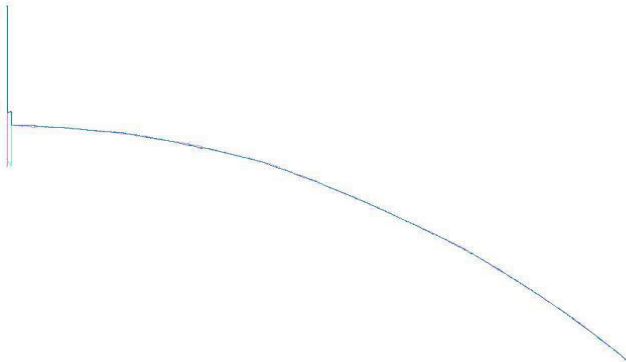


Figure 47.3: 4 Similar Phase Var (units: 1×10^{-7} s) vs Time (0 - 8) d

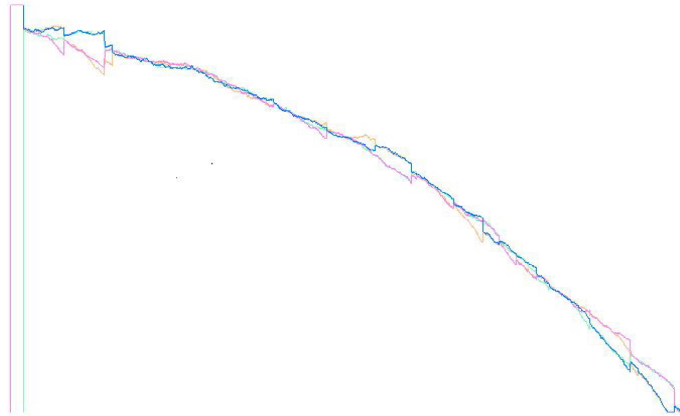


Figure 47.4: 4 Similar Phase Var (units: 1×10^{-8} s) vs Time (0 - 3) d

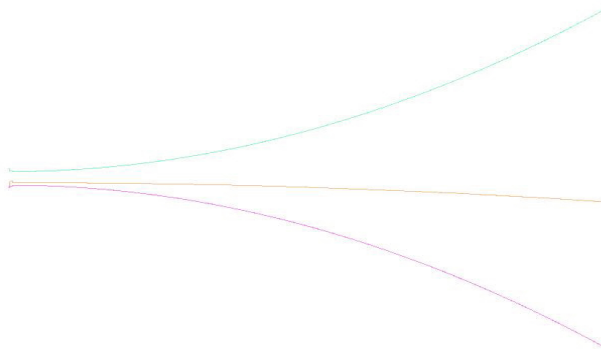


Figure 47.5: S1 Phase Var & Cov Bounds (units: 1×10^{-6} s) vs Time (0 - 8) d

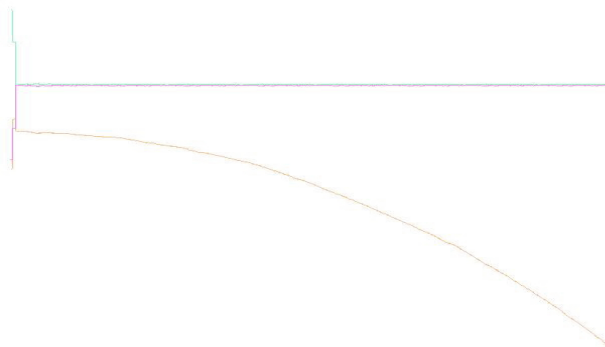


Figure 47.6: S1 Phase Var Reduced Cov (units: 1×10^{-7} s) vs Time (0 - 8) d

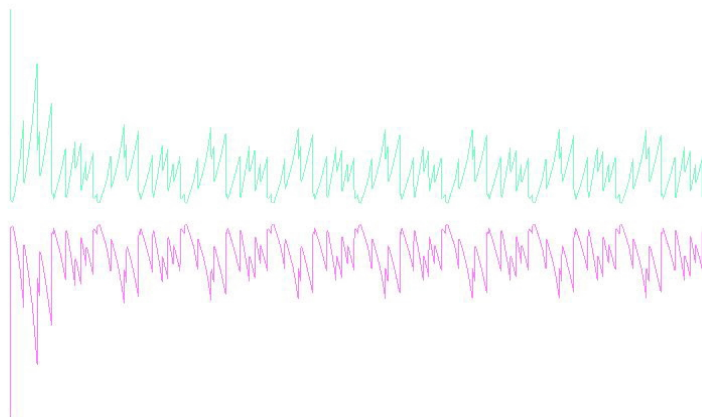


Figure 47.7: S1 Phase Var Reduced Cov (units: 1×10^{-9} s) vs Time (0 - 8) d

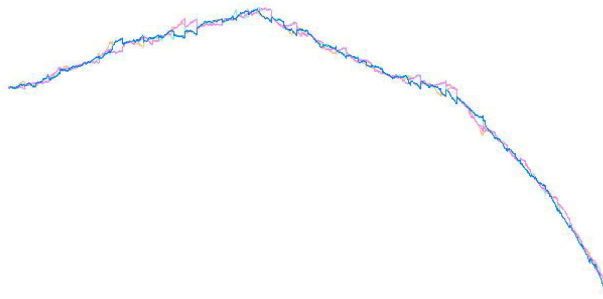


Figure 47.8: 4 Similar Phase Var no IC (units: 1×10^{-8} s) vs Time (0 - 8) d

Part XIII

Appendices

A. The Least Squares Quadratic

Given:

$$\tilde{J}(\Delta X_0) = (\Delta y - A\Delta X_0)^T W (\Delta y - A\Delta X_0)$$

one can derive:

$$\frac{\partial \tilde{J}}{\partial \Delta X_0} = 2(A^T W A \Delta X_0 - A^T W \Delta y)$$

by differentiating $\tilde{J}(\Delta X_0)$ as though its elements were scalars but inserting the matrix transpose operator to obtain consistent matrix dimensionality. Note first that:

$$\frac{\partial \Delta y}{\partial \Delta X_0^T} = 0_{m \times n}$$

and:

$$\frac{\partial \Delta X_0}{\partial \Delta X_0^T} = I_{n \times n}.$$

Then:

$$\begin{aligned} \frac{\partial \tilde{J}}{\partial \Delta X_0} &= \left(\frac{\partial \Delta y}{\partial \Delta X_0^T} - A \frac{\partial \Delta X_0}{\partial \Delta X_0^T} \right)^T W (\Delta y - A\Delta X_0) + \left[(\Delta y - A\Delta X_0)^T W \left(\frac{\partial \Delta y}{\partial \Delta X_0^T} - A \frac{\partial \Delta X_0}{\partial \Delta X_0^T} \right) \right]^T \\ &= -A^T W \Delta y + A^T W A \Delta X_0 + [-\Delta y^T W A + \Delta X_0^T A^T W A]^T \\ &= 2(A^T W A \Delta X_0 - A^T W \Delta y). \end{aligned}$$

B. Sequential Least Squares

The *sequential* least squares algorithm derives from the batch least squares algorithm, and is essentially equivalent to it given the least squares hypothesis.

We are given an a priori state estimate $n \times 1$ matrix $\hat{X}_{0|0}$ and m measurements y_j , where $j \in \{1, 2, \dots, m\}$. The iterative least squares normal equation is given by Eq. 36.9:

$$(A^T W A) \Delta \hat{X}_{0|m} = A^T W \Delta \hat{y} \quad (1)$$

where matrix A is an $m \times n$ ($m > n$) matrix, W is an $m \times m$ diagonal matrix of measurement weights, $\Delta \hat{X}_{0|m}$ is an $n \times 1$ state correction estimate at time t_0 , and $\Delta \hat{y}$ is an $m \times 1$ matrix of measurement residuals. The subscript notation $0|m$ denotes epoch at time t_0 , and the inclusion of measurements through time t_m .

Eq. 1 can also be written:

$$\left(\sum_{j=1}^m A_j^T W_j A_j \right) \Delta \hat{X}_{0|m} = \left(\sum_{j=1}^m A_j^T W_j \Delta \hat{y}_j \right) \quad (2)$$

To see this, insert:

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_m \end{bmatrix}$$

$$W = \begin{bmatrix} W_1 & & & 0 \\ & W_2 & & \\ & & \ddots & \\ 0 & & & W_m \end{bmatrix}$$

$$\Delta \hat{y} = \begin{bmatrix} \Delta \hat{y}_1 \\ \Delta \hat{y}_2 \\ \vdots \\ \Delta \hat{y}_m \end{bmatrix}$$

into Eq. 1. A_j is 1 x n, W_j is 1 x 1, and:

$$\Delta \hat{y}_j = y_j - y(\hat{X}_{j|0}) \quad (3)$$

is 1 x 1. y_j is a measurement value, and the measurement representation $y(\hat{X}_{j|0})$ is a nonlinear function of $\hat{X}_{j|0}$. Measurement weight W_j is the reciprocal of white noise measurement error variance $\sigma_{y,j}^2$:

$$\sigma_{y,j}^2 = 1/W_j. \quad (4)$$

If the $n \times n$ matrix $A^T W A$ is non-singular (has no zero eigenvalues), then it is inverted to produce the state correction:

$$\Delta \hat{X}_{0|m} = (A^T W A)^{-1} (A^T W \Delta \hat{y}) \quad (5)$$

The correction $\Delta \hat{X}_{0|m}$ is added to the a priori state estimate $\hat{X}_{0|0}$ at time t_0 , and Eq. 5 is iterated recursively until the sum of squares of weighted measurement residuals $\Delta \hat{y}^T W \Delta \hat{y}$ (weighted RMS squared) is less than some preassigned positive small number, or until the iterative relative weighted RMS magnitude on measurement residuals is less than some preassigned positive number. It is important to note that the least squares algorithm is derived by minimizing $\Delta \hat{y}^T W \Delta \hat{y}$.

Matrix A_j is constructed from the product:

where y_j is the measurement representation for time t_j , and the linear state error transition matrix $\Phi_{j,0}$ moves state information from epoch at time t_0 to time of measurement t_j .

It is well known by users of batch least squares orbit determination that the best accuracy prediction performance is obtained empirically by finding a batch time interval which is long enough to capture sufficient measurement information, and is short enough to avoid the effects of random force modeling errors. Suppose here that m measurements requires an excessive batch measurement time interval, and that p < m measurements does satisfy the time interval requirement.

Partition the batch of m measurements into two batches of sizes p and q: m = p + q. Then $t_q = t_m$ and equation 2 can be written:

$$\left(\sum_{j=1}^p A_j^T W_j A_j + \sum_{j=p+1}^{p+q} A_j^T W_j A_j \right) \Delta \hat{X}_{0|q} = \left(\sum_{j=1}^p A_j^T W_j \Delta y_j + \sum_{j=p+1}^{p+q} A_j^T W_j \Delta y_j \right). \quad (6)$$

Now suppose in real time that the last measurement we have received is at time t_p . We do not yet have the measurement at time t_{p+1} . The least squares normal equation can be written:

$$\left(\sum_{j=1}^p A_j^T W_j A_j \right) \Delta \hat{X}_{0|p} = \left(\sum_{j=1}^p A_j^T W_j \Delta y_j \right), \quad (7)$$

similar to Eq. 2. The state correction estimate can then be calculated:

$$\Delta \hat{X}_{0|p} = \left(\sum_{j=1}^p A_j^T W_j A_j \right)^{-1} \left(\sum_{j=1}^p A_j^T W_j \Delta y_j \right) \quad (8)$$

together with the least squares state error covariance matrix:

$$P_{0|p} = \left(\sum_{j=1}^p A_j^T W_j A_j \right)^{-1} \quad (9)$$

if the information matrix $\left(\sum_{j=1}^p A_j^T W_j A_j \right)$ is non-singular. Iterative differential corrections are performed as defined above. Now insert Eq. 9 into Eq. 6:

$$\left(P_{0|p}^{-1} + \sum_{j=p+1}^{p+q} A_j^T W_j A_j \right) \Delta \hat{X}_{0|q} = \left(\sum_{j=1}^p A_j^T W_j \Delta y_j + \sum_{j=p+1}^{p+q} A_j^T W_j \Delta y_j \right). \quad (10)$$

Suppose we have just received the second batch of q measurements. How shall we use Eq. 10? Note that the $n \times 1$ matrix $\sum_{j=1}^p A_j^T W_j \Delta y_j$ is presumably negligible due to minimizing $\Delta y^T W \Delta y$ over the first batch of p measurements. Then Eq. 10 can be written:

$$\left(P_{0|p}^{-1} + \sum_{j=p+1}^{p+q} A_j^T W_j A_j \right) \Delta \hat{X}_{0|q} = \left(\sum_{j=p+1}^{p+q} A_j^T W_j \Delta y_j \right) \quad (11)$$

and the correction:

$$\Delta \hat{X}_{0|q} = \left(P_{0|p}^{-1} + \sum_{j=p+1}^{p+q} A_j^T W_j A_j \right)^{-1} \left(\sum_{j=p+1}^{p+q} A_j^T W_j \Delta y_j \right) \quad (12)$$

can be calculated together with the least squares covariance matrix:

$$P_{0|q} = \left(P_{0|p}^{-1} + \sum_{j=p+1}^{p+q} A_j^T W_j A_j \right)^{-1} \quad (13)$$

if the least squares information matrix $\left(P_{0|p}^{-1} + \sum_{j=p+1}^{p+q} A_j^T W_j A_j \right)$ is non-singular. Given the least squares hypothesis, we note that $\Delta \hat{X}_{0|q}$ is essentially the same correction obtained in Eq. 5, but here we have obtained it sequentially in two measurement batches rather than one.

Important: As Jim Woodburn emphatically notes, $\Delta \hat{X}_{0|q}$ is not the same correction obtained in Eq. 5 when processing *real data*. This is due to existence

of force modeling errors and the nonlinearity produced by iterative least squares corrections.

Recall from above we have assumed that the simultaneous processing of m measurements requires an excessive batch measurement time interval $[t_1, t_m]$; *i.e.*, random force modeling errors are hurting us. Then Eq. 13 needs to be modified to decrease the influence of the a priori information matrix $P_{0|p}^{-1} = \left(\sum_{j=1}^p A_j^T W_j A_j \right)$ on the state error correction estimate $\Delta \hat{X}_{0|q}$ of Eq. 12. This is equivalent to increasing variances of the a priori covariance matrix $P_{0|p} = \left(\sum_{j=1}^p A_j^T W_j A_j \right)^{-1}$. When this is performed the a priori information matrix $P_{0|p}^{-1}$ is said to be deweighted and the batch least squares algorithm is transformed into a batch filter.

C. Ad-Hoc Batch Filter

We have derived the batch least squares algorithm and the sequential least squares algorithm above. The batch filter algorithm is derived below, ad hoc, from the sequential least squares algorithm.

Swerling [104] derived a version of the batch filter from the maximum likelihood quadratic form in 1958. This unitless quadratic form J has two terms: $J = \Delta y^T W \Delta y + \Delta X_0^T P_0^{-1} \Delta X_0$. The first term is the *least squares* quadratic form and the second term is a quadratic form on the difference ΔX_0 between the previous state estimate and the new state estimate, both at fixed epoch t_0 , and weighted by an a priori positive definite state error covariance matrix P_0 . The maximum likelihood algorithm, derived by minimization of J , can be viewed as minimizing the sum of squares of measurement residuals while simultaneously constraining the new state estimate to the previous state estimate. The batch filter is an intermediate method that has characteristics of both a least squares algorithm and a filter. From a user perspective, the maximum likelihood algorithm is distinguished from least squares in that it enables the specification of an a priori state error covariance matrix to constrain the new state estimate to the previous state estimate. The user typically specifies a positive diagonal a priori covariance matrix, with *large* values to free the new state estimate from the a priori state estimate, and/or *small* values to bind the new state estimate to the a priori state estimate. Swerling did not discover Schur's matrix identities (inversion lemma, see Meditch [77]), and was thereby blocked from removing the debilitating $n \times n$ matrix inverse with state size n , associated with least squares and the batch filter. The inversion lemma is required to convert from simultaneous measurement batch processing to sequential processing without the $n \times n$ matrix inverse. Sequential processing is a necessary condition for optimal estimation. It was left to Kalman to discover Sherman's theorem and thereby introduce a *stochastic* performance functional for optimal estimation.

Recall Eq. 35.11, and define:

$$\Phi_{j,i} = \frac{\partial X_j}{\partial X_i} \quad (14)$$

for any times t_i and t_j . Insert $A_j = H_j \Phi_{j,0}$ and $W_j = R_j^{-1}$ (recall Eqs. 35.13 and ??) into Eqs. 9, 12 and 13 to get:

$$P_{0|p} = \left(\sum_{j=1}^p (H_j \Phi_{j,0})^T R_j^{-1} (H_j \Phi_{j,0}) \right)^{-1} \quad (15)$$

$$\Delta \hat{X}_{0|q} = P_{0|q} \left(\sum_{j=p+1}^{p+q} (H_j \Phi_{j,0})^T R_j^{-1} \Delta y_j \right) \quad (16)$$

$$P_{0|q} = \left(P_{0|p}^{-1} + \sum_{j=p+1}^{p+q} (H_j \Phi_{j,0})^T R_j^{-1} (H_j \Phi_{j,0}) \right)^{-1}. \quad (17)$$

Our least squares epoch was arbitrarily selected to be time t_0 . We could just as easily have selected time t_q for epoch. In this case the above three equations become:

$$P_{q|p} = \left(\sum_{j=1}^p (H_j \Phi_{j,q})^T R_j^{-1} (H_j \Phi_{j,q}) \right)^{-1} \quad (18)$$

$$\Delta \hat{X}_{q|q} = P_{q|q} \left(\sum_{j=p+1}^{p+q} (H_j \Phi_{j,q})^T R_j^{-1} \Delta y_j \right) \quad (19)$$

and:

$$P_{q|q} = \left(P_{q|p}^{-1} + \sum_{j=p+1}^{p+q} (H_j \Phi_{j,q})^T R_j^{-1} (H_j \Phi_{j,q}) \right)^{-1}. \quad (20)$$

Premultiply Eq 18 by $\Phi_{p,q}$, post multiply it by $\Phi_{p,q}^T$, and use the transitive property $\Phi_{r,p} = \Phi_{r,q} \Phi_{q,p}$ with $r = p$ and $\Phi_{p,p} = I$ to get $\Phi_{p,q} = \Phi_{q,p}^{-1}$ and:

$$P_{p|p} = \left(\sum_{j=1}^p (H_j \Phi_{j,p})^T R_j^{-1} (H_j \Phi_{j,p}) \right)^{-1} \quad (21)$$

to see that Eq. 18 can be written:

$$P_{q|p} = \Phi_{q,p} P_{p|p} \Phi_{q,p}^T. \quad (22)$$

Eqs. 19, 20, 22, and 21 define the exact same sequential least squares algorithm derived in Appendix B, but with different epoch placement. With these changes in epoch placement, the last paragraph of Appendix B reads: Then Eq. 22 needs to be modified to decrease the influence of the a priori information matrix $P_{p|p}^{-1}$ on the state error correction estimate $\Delta \hat{X}_{q|q}$ of Eq. 19. This is equivalent to increasing variances of the a priori covariance matrix $P_{p|p}$, embedded in Eq. 22.

Thus the conversion of this sequential least squares algorithm to a batch filter is achieved by replacing, ad hoc, covariance matrix $P_{q|p} = \Phi_{q,p} P_{p|p} \Phi_{q,p}^T$ with:

$$P_{q|p} = \Phi_{q,p} P_{p|p} \Phi_{q,p}^T + Q_{q,p}. \quad (23)$$

For each measurement batch recursion, the batch filter algorithm execution consists of a time update and a measurement update. Assume that we have just finished processing the measurement batch at time t_p , and that the next measurement batch time is $t_q > t_p$. Given the state estimate $\hat{X}_{p|p}$ at time t_p , it is numerically integrated to time t_q to obtain $\hat{X}_{q|p}$, the *state estimate time update*. Then the state error *covariance time update* is defined by Eq. 23. Then the state error *covariance measurement update* is defined by Eq 20. Then the *state estimate measurement update* is defined by Eq. 19 and:

$$\hat{X}_{q|q} = \hat{X}_{q|p} + \Delta \hat{X}_{q|q}. \quad (24)$$

Given the next measurement batch time at $t_r > t_q$, then repeat the above procedure with $p \rightarrow q$ and $q \rightarrow r$.

The sums $\sum_{j=p+1}^{p+q} (H_j \Phi_{j,q})^T R_j^{-1} \Delta y_j$ and $\sum_{j=p+1}^{p+q} (H_j \Phi_{j,q})^T R_j^{-1} (H_j \Phi_{j,q})$ of Eqs. 19 and 20 are processed simultaneously, they define the measurement batch in "batch filter". Simultaneous batch processing is obviously faster than individual measurement processing.

It must be noted that any *optimal physically connected sequential filter* processes each measurement separately, folding in measurement information Δy_j and force modeling error covariance Q_j sequentially as they occur. The optimal real time filter does *not* process any batch of measurements simultaneously, for this destroys both optimal estimate accuracy and the autonomous measurement residual editor.

Finally, given a rigorous derivation of an *optimal physically connected sequential filter* from the fundamental theorem of estimation theory [77], one can "derive" the batch filter from it with a degrading ad-hoc replacement: Arbitrarily replace any sequence of measurement/deweighting calculations with a simultaneous batch calculation (thereby ignoring all random force modeling errors between measurement times) and retain deweighting at times between batches.

D. Kalman's Model Equation

Calculus provides the definition:

$$\frac{dX}{dt} = \lim_{\Delta t \rightarrow 0} \frac{X(t + \Delta t) - X(t)}{\Delta t} = \lim_{\Delta t \rightarrow 0} \frac{\Delta X}{\Delta t} \quad (25)$$

Thus the deterministic linear differential equation:

$$\frac{dX}{dt} = A(t)X + B(t)\omega \quad (26)$$

can be written:

$$\lim_{\Delta t \rightarrow 0} \frac{\Delta X}{\Delta t} = A(t)X + B(t)\omega \quad (27)$$

Undo the calculus (remove the limit) to write:

$$\Delta X = A(t)X\Delta t + B(t)\omega\Delta t \quad (28)$$

Transform this deterministic equation to a stochastic equation with the white noise assignment:

$$\omega = \frac{\Delta\beta}{\Delta t} \quad (29)$$

according to Eq. 13.95, where β is Brownian motion. Thus:

$$\Delta X = A(t)X\Delta t + B(t)\Delta\beta \quad (30)$$

provides a well-defined linear ordinary stochastic differential equation where variations ΔX in the state are driven by independent increments (white noise) $\Delta\beta$ of Brownian motion. See Bucy [10] page 24 Eq. 2.5 for comparison.

.0.5 Measurements at $t_j = t_{k+1}, t_{k+2}, \dots$

In the section *Measurement at $t_j = t_{k+1}$* above, replace t_{k+1} with t_{k+2} for the measurement $y_j = y_{k+2}$ at time $t_j = t_{k+2}$. When $t_j = t_{k+h}$, replace t_{k+1} with t_{k+h} for the measurement $y_j = y_{k+h}$ at time $t_j = t_{k+h}$.

Bibliography

- [1] Allan, D.W., *Time and Frequency Characterization, Estimation, and Prediction of Precision Clocks and Oscillators*, NIST Technical Note 1337, 1990
- [2] Antreasian, Peter Garo, *Precision Radiation Force Modeling for the TOPEX/POSEIDON Mission*, Phd Thesis, University of Colorado, 1992
- [3]
- [4] Baker, Robert M.L., *Astrodynamics: Applications and Advanced Topics*, Academic Press, New York, 1967
- [5] Bancroft, Stephen, "An Algebraic Solution of the GPS Equations." IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-21, No. 7, January, 1985.
- [6] Bar-Sever, Y., Kuang, D., "New Empirically Derived Solar Radiation Pressure Model for Global Positioning System Satellites", IPN Progress Report 42-159, November 15, 2004.
- [7] Bar-Sever, Y., Kuang, D., "New Empirically Derived Solar Radiation Pressure Model for Global Positioning System Satellites During Eclipse Seasons", IPN Progress Report 42-160, February 15, 2005.
- [8] Adi Ben-Israel, Thomas N. E. Greville, *Generalized Inverses: Theory and Application*, Wiley, New York, 1974
- [9] Gerald J Bierman, *Factorization Methods for Discrete Sequential Estimation*, Academic Press, New York, 1977
- [10] Bucy, Richard S., Joseph, Peter D., *Filtering for Stochastic Processes with Applications to Guidance*, Interscience Publishers, John Wiley & Sons, New York, 1968
- [11] Brown, Kenneth R., *The Theory of the GPS Composite Clock*, Proc. Inst. Navig.– Global Posit. Syst., 1991, pp. 223-241
- [12] Budden, K., G., *The Propagation of Radio Waves*, Cambridge University Press, England, 1972

- [13] The Committee for the COSPAR International Reference Atmosphere (CIRA) of COSPAR Working Group 4, Committee On Space Research (COSPAR), *CIRA 1972*, Akademie-Verlag, Berlin, 1972
- [14] Collins, J. Paul, An Overview of GPS Inter-Frequency Carrier Phase Combinations, UNB/GSD, October 1999 (WEB)
- [15] R. K. Crane, M. L. Meeks, "Refraction Effects in the Neutral Atmosphere", *Methods of Experimental Physics: Astrophysics 12B*, Academic Press, 1976
- [16] Davenport, Wilbur B., Root, William L., *An Introduction to Random Signals and Noise*, McGraw-Hill, New York, 1958
- [17] J. L. Davis, T. A. Herring, I. I. Shapiro, A. E. E. Rogers, G. Elgered, *Geodesy by Radio Interferometry: Effects of Atmospheric Modeling Errors on Estimates of Baseline Length*, Radio Science, vol. 20, 1985.
- [18] Daniel N. Lapedes, *McGraw-Hill Dictionary of Physics and Mathematics*, McGraw-Hill Book Company, New York, 1978
- [19] Doob, J. L., *Stochastic Processes*, John Wiley & Sons, New York, 1953.
- [20] Paul Dyer and Stephen McReynolds, *Extension of Square Root Filtering to Include Process Noise*, J. Ops. Theory and Appl. 3, No. 6, 444-459, 1969
- [21] Feller, W., *An Introduction to Probability Theory and its Applications*, Vol. 1, John Wiley & Sons, New York, 1950.
- [22] Feller, W., *An Introduction to Probability Theory and its Applications*, Vol. 2, John Wiley & Sons, New York, 1970, pp. 262-3.
- [23] Fliegel, H.F., Gallini, T.E., Swift, E.R., "Global Positioning System Radiation Force Model for Geodetic Applications", Journal of Geophysical Research, Volume 97, No. B1, January 1992.
- [24] Fliegel, H.F., Gallini, T.E., "Solar Force Modeling of Block IIR Global Positioning System Satellites", Journal of Spacecraft and Rockets, Volume 33, No. 6, November-December 1996.
- [25] Foster, J., Richards, F. B., *The Gibbs Phenomenon for Piecewise-Linear Approximations*, The American Mathematical Monthly, Vol. 98, No. 1. Jan., 1991, pp. 47-49
- [26] Gauss, K. F., *Theoria Motus Corporum Coelestium (Theory of the Motion of the Heavenly Bodies Moving about the Sun in Conic Sections)*, Little, Brown, and Co., 1857, republished by Dover, 1963
- [27] J. Willard Gibbs, *Vector Analysis*, Dover Edition, 1960, from Charles Scribner's & Sons, 1909

- [28] Paul Glasserman, *Monte Carlo Methods in Financial Engineering*, Springer, 2004
- [29] Gene H. Golub, Charles F. Van Loan, *Matrix Computations*, Johns Hopkins Press, 1989
- [30] Gooding, R. H., *A New Procedure for Orbit Determination Based on Three Lines of Sight (Angles Only)*, Technical Report 93004, Defense Research Agency (GU14 6TD UK), Farnborough, Hampshire, April 1993
- [31] Gooding, R. H., *On the Solution of Lambert's Orbital Boundary-Value Problem*, Technical Report 88027, Royal Aerospace Establishment, Procurement Executive, Minister of Defence, Farnborough, Hants, 1988
- [32] *GEONS Time Bias Propagation*, GSFC, FDSS-14-0006, pages 4-36 through 4-39
- [33] Herrick, Samuel, *Astroynamics*, Vol 1, Van Nostrand, London, 1971
- [34] Herrick, Samuel, *Astroynamics*, Vol 2, Van Nostrand, London, 1972
- [35] T. A. Herring, *Modeling Atmospheric Delays in the Analysis of Space Geodetic Data*, Proceedings of the Symposium on Refraction of Transatmospheric Signals in Geodesy, Netherlands, Delft, No. 36, New Series, pp 157-164, 1992.
- [36] Ho, Y.C., *The Method of Least Squares and Optimal Filtering Theory*, Rand Corporation, Oct 1962
- [37] D. B. Holdridge, *An Alternate Expression for Light Time Using General Relativity*, JPL Space Program Summary 37-48, III, pp. 2-4. This reference is given in McCarthy[71].
- [38] A. S. Householder, *Unitary Triangularization of a Nonsymmetric Matrix*, J. Ass. Comp. Mach. 5, 205-243, 1958
- [39] Hughes, Peter C., *Spacecraft Attitude Dynamics*, John Wiley & Sons, 1988. pp. 522-34.
- [40] Hujsak, R.S., *Solar Pressure*, Proceedings of the Artificial Satellite Theory Workshop, Edited by P. Kenneth Seidelmann and Bernard Kaufman, U.S. Naval Observatory, Washington DC, Nov 8,9, 1993
- [41] GPS NAVSTAR JPO, *ICD-GPS-200, REV. C, 10 October 1993, NAVSTAR GPS Space Segment/Navigation User Interface, IRN-200C-004*, 12 April 2000
- [42] GPS NAVSTAR JPO, *ICD-GPS-208, 5 June 1987, Auxiliary User Interfaces*

- [43] I. M. Ifadis, *Space to Earth Geodetic Observations: Approaching the Atmospheric Effect*, Aristotle University, (1999 - latest reference)
- [44] I. M. Ifadis, *A New Approach to Mapping the Atmospheric Effect for GPS Observations*, Aristotle University, Earth Planets Space, 52, 703-708, 2000
- [45] I. Ifadis, *The Atmospheric Delay of Radio Waves: Modelling the Elevation Dependence on a Global Scale*, Licentiate Thesis, Technical Report No. 38L, Gothburg, Sweden, 1986
- [46] I. Ifadis, Contribution to the Study of the Atmospheric Refraction, on Radio Waves Used in Modern Geodetic Techniques in Long Distance Measurements, Thesis Report, vol. IA'6, School of Engineering, Faculty of Civil Engineering, Thessalonika, 1987.
- [47] Eugene Jahnke, Fritz Emde, *Tables of Functions with Formulae and Curves*, Dover, New York, 1945
- [48] Johnson, N.J., S. Kotz, N. Balakrishnan (1994), *Continuous Univariate Distributions*, Vol. 1, 2nd ed., John Wiley & Sons, New York. p. 91.
- [49] Kalman, R. E., *New Methods in Wiener Filtering Theory*, Proceedings of the First Symposium on Engineering Applications of Random Function Theory and Probability, edited by J. L. Bogdanoff and F. Kozin, John Wiley & Sons, New York, 1963.
- [50] Kaplan, Elliott D. (editor), Ward, Phillip (Chapter 5), *Understanding GPS*, Artech House, Boston, 1996E.
- [51] Kaula, William M., *Theory of Satellite Geodesy*, Blaisdell Pub., Mass., 1966
- [52] Kaula, William M., *Statistical and Harmonic Analysis of Gravity*, Journal of Geophysical Research, Vol. 64, Dec. 1959, pp. 2411, 2412, 2418.
- [53] King-Hele, Desmond, *Theory of Satellite Orbits in an Atmosphere*, Butterworths, London, 1964
- [54] Klobuchar, J. A., *The Handbook of Geophysics and the Space Environment*, Air Force Geophysics Laboratory, 1985
- [55] Konopliv, A.S., S.W. Asmar, E. Carranza, W.L. Sjogren, and D.N. Yuan (2001), "Recent Gravity Models as a Result of the Lunar Prospector Mission", *Icarus*, Volume 150, Issue 1, March 2001, Pages 1-18.
- [56] Paul E. Koskela (L. G. Walters, C. G. Hilton, J. L. Arsenault, R. G. Schinnerer, Chela Varrentzoff, K. C. Ford), *Astrodynamic Analysis for the Advanced Orbit/Ephemeris Subsystem*, Aeronutronic Pub. No. U-4180, DDI Subcontract No. 66-1, AF 04(695)-976, Philco-Ford Aeronutronic Div, 1 Sep., 1967

- [57] Kursinski et. al., *The GPS Radio Occultation Technique*, TAO, Vol. 11, No. 1, 53-114, March, 2000.
- [58] Lancaster, E. R., Blanchard, R. C., A Unified Form of Lambert's Theorem, NASA TN D-5368, Sept. 1969
- [59] Richard B. Langley, Wendy Wells, Virgilio de Brito Mendes, *Tropospheric Propagation Delay: a Bibliography*, Geodetic Research Library, Department of Geodesy and Geomatics Engineering, University of New Brunswick, 2nd Edition, 22 April 1995
- [60] Harry Lass, *Vector and Tensor Analysis*, McGraw-Hill, New York, 1950
- [61] Larson, Kristine, Levine, Judah, *Time Transfer Using the Phase of the GPS Carrier*, IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol. 46, No. 4, July 1999
- [62] Larson, Kristine, Levine, Judah, *Carrier Phase Time Transfer*, IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol. 47, No. 2, March 2000
- [63] Link, F., *Lunar Eclipses*, Physics and Astronomy of the Moon, edited by Kopal, Academic Press, New York, 1962
- [64] Larson, Kristine, Levine, Judah, Nelson, Lisa M., Parker, Thomas E., *Assessment of GPS Carrier-Phase for Time-Transfer Applications*, IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol. 46, No. 4, July 1999
- [65] Larson, Kristine, Private Communication (e-mail), 9 July 2007
- [66] Charles L. Lawson, Richard J. Hanson, *Solving Least Squares Problems*, Prentice-Hall, 1974
- [67] Robert G. Lorenz (Assignee: Ashtech Telesis, Inc.), *GPS Commercial Receiver Patent Number 5,134,407*, U. S. Patent Documents, 28 July 1992.
- [68] Robert G. Lorenz (Assignee: Ashtech Telesis, Inc.), *GPS Commercial Receiver Patent Number 5,293,170*, U. S. Patent Documents, 8 March 1994.
- [69] J. W. Marini, *Correction of Satellite Tracking Data for an arbitrary Atmospheric Profile*, Radio Science, vol. 7, pp223-231, 1972.
- [70] McCarthy, D., *IERS Standards*, *IERS Technical Note 13*, 1992
- [71] Dennis D. McCarthy, *IERS Technical Note 21*, US Naval Observatory, July 1996. See Eq. 17 therein.
- [72] McCarthy, Dennis D., *IERS TECHNICAL NOTE 32*, US Naval Observatory, Nov., 2003

- [73] Stephen McReynolds, *Private Communications*, 1980 to 1998.
- [74] Stephen McReynolds, *Editing Data Using Sequential Smoothing Techniques for Discrete Systems*, AIAA/AAS Astrodynamics Conference, August 20-22, 1984, Seattle, WA.
- [75] Stephen McReynolds, *Filter-Smoother Consistency Test*, Interoffice Memo, Martin Marietta, 21 Feb., 1995
- [76] McReynolds, S.R. "Multipoint Smoothing Algorithm Systems", *Journal of Guidance, Control, and Dynamics*, Vol. 12, Number 6, Nov-Dec 1989, pages 920-924.
- [77] Meditch, J. S., *Stochastic Optimal Linear Estimation and Control*, McGraw-Hill, New York, 1969.
- [78] Meditch, J. S., *Orthogonal Projection and Discrete Optimal Linear Smoothing*, SIAM J. Control, vol. 5, p. 74, 1967
- [79] Meditch, J. S., *A Survey of Data Smoothing for Linear and Nonlinear Dynamic Systems*, Automatica, Vol. 9, pp. 151-162, 1973
- [80] Meditch, J. S., Personal Communications, The Aerospace Corporation, 1974.
- [81] Meditch, J. S., Personal Communications, 1974.
- [82] Thomas K. Meehan, Jess Brooks, Lawrence E. Young (Assignee: California Institute of Technology), *GPS Receiver Patent Number 6,061,390*, U. S. Patent Documents, 9 May 2000.
- [83] V. B. Mendes, R. B. Langley, *A Comprehensive Analysis of Mapping Functions Used in Modeling Tropospheric Propagation Delay in Space Geodetic Data*, Paper at KIS94, Banff, Aug 30 – Sep 2, 1994
- [84] C. Moller, *The Theory of Relativity*, Oxford University Press, Delhi, 1952, Clarendon Press 1972
- [85] Anderson, Brian D. O., Moore, John B., *Optimal Filtering*, Prentice-Hall, New Jersey, 1979.
- [86] Oliver Montenbruck, Remco Kroes, *In-Flight Performance Analysis of the CHAMP Blackjack GPS Receiver*, Rev. 1, 31 March 2003.
- [87] Moyer, Theodore D., "Formulation for Observed and Computed Values of Deep Space Network Data Types for Navigation", Monograph 2, Deep Space Communications and Navigation Series.
- [88] A. E. Niell, *Improved Global Atmospheric Mapping Functions for VLBI and GPS*, URSI/IAU Symposium on VLBI Technology – Progress and Future Observational Possibilities, Kyoto, Japan, 6-10 September 1993.

- [89] A. E. Niell, *Global Mapping Functions for the Atmospheric Delay at Radio Wavelengths*, J. Geophys. Res., 101(B2), 3227-3276, 1996
- [90] A. Niell, *The IMF Mapping Functions*, 15 Aug 2002. This paper was downloaded from the WEB.
- [91] O'Toole, James W., "Mathematical Description of the OMNIS Satellite Orbit Generation Program (OrbGen)", NSWCDD/TR-02/118, May 2004.
- [92] J. S. Owens, *Optical Refractive Index of Air: Dependence on Pressure, Temperature, and Composition*, Applied Optics, vol. 6, No. 1, pp 51-59, 1967.
- [93] Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, 1965.
- [94] Parker, Thomas E., Private Communication (Geneva), 30 May 2007
- [95] Pechenick, K. R., *Solar Radiation Pressure on Satellites and Other Related Effects*, The General Electric Company, 1983
- [96] Pechenick, K. R., *A Derivation of the Gravity Error Covariance Formulas Including the Off-Diagonal Components*, Applied Technology Associates (ATA), GSFC Contract No. NAS5-29417, September 1988
- [97] Rauch, H. E., *Solutions to the Linear Smoothing Problem*, IEEE Trans. Autom. Control, vol. AC-8, p. 371, 1963
- [98] Rauch, H. E., Tung, F., Striebel, C. T., *Maximum Likelihood Estimates of Linear Dynamical Systems*, AIAA J., vol 3, p. 1445, 1965
- [99] Davenport, Wilbur B., Root, William L., *An Introduction to the Theory of Random Signals and Noise*, McGraw-Hill, New York, 1958.
- [100] Schmidt, Stanley F., *Compensation for Modeling Errors in Orbit Determination Problems (Interim Report)*, Report No. 67-16, Contract NAS 5-11048, November 1967, Analytical Mechanics Associates, Inc., 57 Old Country Rd, Westbury, NY, 11590
- [101] Sherman, S., *A Theorem on Convex Sets with Applications*, Ann. Math. Stat., 26, 763-767, 1955.
- [102] Sherman, S., *Non-Mean-Square Error Criteria*, IRE Transactions on Information Theory, Vol. IT-4, 1958.
- [103] Digital Systems Section Code 531, *Tracking and Acquisition Handbook for the Spacecraft Tracking and Data Network*, 450-TAH-STDN, October 1994, Section 4.2.2.3 Data Reduction Algorithms, subsection c. Observed Range-Rate.

- [104] Swerling, P. *A Proposed Stageswise Differential Correction Procedure for Satellite Tracking and Prediction*, P-1292, Rand Corporation, 8 Jan 1958
- [105] Zucca, Cristina; Tavella, Patricia, *The Clock Model and Its Relationship with the Allan and Related Variances*, IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol. 52, No. 2, Feb. 2005
- [106] A. M. Smith, *REFRACTION CORRECTION*, RC, TM-(L)-5048/492/01, System Development Corporation, 11 August, 1978
- [107] Phung, P. B., Guedeney, V. S., Teles, J., *Tracking and Data Relay Satellite System (TRDSS) Range and Doppler Tracking System Observation Measurement and Modeling*, X-572-80-26, NASA GSFC, Sept. 1980, Equation 4-2 and its derivation.
- [108] Thayer, G. D., *An Improved Equation for the Radio Refractive Index of Air*, Radio Science, vol. 9, pp. 803-807, 1974.
- [109] Thode, Henry C. Jr. *Testing for Normality*, CRC Press, 2002.
- [110] U.S. Air Force by Boeing Navigation Systems, GPS Operational Control System Mathematical Algorithms Document, Volume GOMA-E, 27 Sep., 2005. Distribution authorized to US DOD and its contractors only.
- [111] O. A. Vasicek, *An Equilibrium Characterization of the Term Structure*, Journal of Financial Economics, 5:177-188
- [112] Vokrouhlicky, D., Farinella, P., Mignard, F., *Solar Radiation Pressure Perturbations for Earth Satellites, I: A Complete Theory Including Penumbra Transitions*, Astronomy and Astrophysics, 17.3, 1993
- [113] Benton, William (publisher), *Webster's Third New International Dictionary of the English Language Unabridged*, Encyclopaedia Britannica, Inc, Chicago, 1966
- [114] Mark L. Wencek, *The Hydrosphere*, www.cmmmp.ucl.ac.uk/~jhh/2B27/water.rtf
- [115] Wiberg, D. M., *Theory and Problems of State Space and Linear Systems*, Schaum's Outline Series, McGraw-Hill, New York, 1971.
- [116] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965
- [117] Jacob Williams, E. Glenn Lightsey, SungPil Yoon, Robert E. Shutz, *Testing of the ICESat Blackjack GPS Receiver Engineering Model*, Center for Space Research, The University of Texas at Austin, downloaded from WEB, no date given.
- [118] Woodburn, J.W., Coppola, V., "Effect of Coordinate Section on Orbit Determination." Paper AAS 13-825, AAS/AIAA Astrodynamics Specialist Conference, August 2013, Hilton Head, SC.

- [119] Wright, J. R., *Sequential Orbit Determination with Auto-Correlated Gravity Modeling Errors*, AIAA, Journal of Guidance and Control, Vol 4, No. 2, May-June 1981, page 304.
- [120] Wright, J. R., *Orbit Determination Solution to the Non-Markov Gravity Error Problem*, AAS/AIAA Paper AAS 94-176, AAS/AIAA Spaceflight Mechanics Meeting, Cocoa Beach, FLA, Feb., 1994
- [121] Wright, James R., *Optimal Orbit Determination*, Paper AAS 02-192, AAS/AIAA Space Flight Mechanics Meeting, San Antonio, Texas, 27-30 Jan., 2002
- [122] Wright, James R., *Real-Time Estimation of Local Atmospheric Density*, Paper AAS 03-164, 13th AAS/AIAA Space Flight Mechanics Meeting, Ponce, Puerto Rico, 9-13 Feb., 2003
- [123] Wright, James R., Woodburn, James, *Simultaneous Real-Time Estimation of Atmospheric Density and Ballistic Coefficient*, Paper AAS 04-175, 14th AAS/AIAA Space Flight Mechanics Conference, Maui, Hawaii, 8-12 Feb., 2004
- [124] Wright, James R., Tanygin, Sergei, *Removal of Arbitrary Discontinuities in Atmospheric Density*, Paper AAS 04-176, 14th AAS/AIAA Space Flight Mechanics Conference, Maui, Hawaii, 8-12 Feb., 2004
- [125] Wright, James R., *Sherman's Theorem*, The Malcolm D. Shuster Astronautics Symposium, The Journal of the Astronautical Sciences, Vol. 54, Nos. 3 and 4, July - December 2006, pp. 299 - 319
- [126] Wright, James R, Woodburn, James, Truong, Son, Chuba, William, *Orbit Gravity Error Covariance*, Paper AAS 08-157, 18th AAS/AIAA Space Flight Mechanics Meeting, Galveston, Texas, January 2008
- [127] Wright, James R, Woodburn, James, Truong, Son, Chuba, William, *Sample Orbit Covariance Function and Filter-Smoother Consistency Tests*, Paper AAS 08-159, 18th AAS/AIAA Space Flight Mechanics Meeting, Galveston, Texas, January 2008
- [128] Wright, James R, Woodburn, James, Truong, Son, Chuba, William, *Orbit Covariance Inner Integrals with Polynomials*, Paper AAS 08-161, 18th AAS/AIAA Space Flight Mechanics Meeting, Galveston, Texas, January 2008
- [129] Wright, James R, *GPS Composite Clock Analysis (two-state clocks)*, Hindawi Publishing Corporation, International Journal of Navigation and Observation, Volume 2008, Article ID 261384
- [130] Wright, James R, *Composite Clocks with Three-State Models*, Paper 33, Thirty-Ninth Annual Precise Time and Time Interval (PTTI) Systems and Applications Meeting, Hyatt Regency, Long Beach, CA, November 2007

- [131] Wright, James R, Woodburn, James, *Nonlinear Variable Lag Smoother*, F. Landis Markley Astronautics Symposium, American Astronautical Society, Cambridge, Maryland, July 2, 2008
- [132] Wu, S. C., *et. al*, Global Positioning System (GPS) Precision Orbit Determination (POD) Software Design, TOPEX/Poseidon Project, JPL D-7275, March 9, 1990.
- [133] Ming YANG and Kuo-Hwa CHEN, "Performance Assessment of a Noniterative Algorithm for GPS Absolute Positioning", Proc. Natl. Sci. Council. ROC(A) Vol. 25, No. 2, 2001, pp. 102-106.